

Kalle Braun · Jürgen Buchmüller · Frank Seger

Das
Colour-Genie
Buch 2

TCS
COMPUTER GMBH

Das Colour-Genie Buch 2

© 1984 TCS Computer GmbH
ISBN 3 88965 004 X

Alle Rechte vorbehalten insbesondere auch diejenigen aus der spezifischen Gestaltung Anordnung und Einteilung des angebotenen Stoffes. Der auszugsweise oder teilweise Nachdruck sowie fotomechanische Wiedergabe oder Übertragung auf Datenträger zur Weiterverarbeitung ist untersagt und wird als Verstoß gegen das Urheberrechtsgesetz und als Verstoß gegen das Gesetz gegen den unlauteren Wettbewerb gerichtlich verfolgt. Für etwaige technische Fehler sowie für die Richtigkeit aller in diesem Buch gemachten Angaben übernehmen der Herausgeber und Autor keine Haftung.

Vorwort

Lieber Colour-Genie Freund !

Nachdem unser 'Colour-Genie-Buch I' ein echter 'Renner' geworden ist, baten uns viele Colour-Genie-Benutzer, ein zweites Buch speziell für ihren Computer herauszubringen.

Gesagt - Gatan:

Vor Ihnen liegt das 'Colour- Genie-Buch II'!

Auch dieses Mal finden Sie auf über 100 Seiten interessante Programme und Tricks; schauen Sie in das Inhaltsverzeichnis auf der nächsten Seite, um sich davon zu überzeugen.

Neben mehreren interessanten Spielen, darunter ein komplett erklärtes Maschinensprachenspiel (!), Graphik- und Musikprogrammen, werden vor allen Dingen die Disk-Benutzer unter Ihnen wissenswertes in diesem Buch finden, z.B. eine schnelle Routine, mit der Disketten formatiert werden können u.v.m.

Wir wünschen Ihnen viel Spaß beim Hacken und sei der Tippfehlerteufel nicht allzusehr gegen Sie!

Bonn, im Dezember 1983

Kalle Braun
Jürgen Buchmüller
Frank Seger



Inhaltsverzeichnis

Vorwort	Seite 1
Einige allgemeine Hinweise	Seite 2
Ein Musikprogramm in Basic - der 'Bluesmaker'	Seite 3
Enterprise - ein strategisches Weltraumspiel	Seite 5
Run to Morlock's Tower - ein Abenteuerspiel	Seite 25
Bar - ein echtes Hilfsprogramm	Seite 42
Ein Maschinensprache-Monitor in Basic	Seite 53
Ein Grafikprogramm - Lissajous-Figuren.....	Seite 58
Zalaga - Ein erläutertes Maschinensprachenspiel.....	Seite 62
Einleitung	Seite 63
Hexadezimal Listing	Seite 64
Disassembliertes Listing mit Kommentaren ...	Seite 72
Der Zeichensatz von Zalaga	Seite 85
Ein schnelles Formatier-Programm	Seite 87
Erklärung des Formatier-Programms	Seite 90
Disassembliertes Listing des Formatier-Programms	Seite 92
Sektoren lesen und schreiben vom Basic aus	Seite 98
Zwei Tips für den Betrieb eines einfarbigen Monitors	Seite 99
Anhang A: Colour-Basic Tokens	Seite 100
Anhang B: Dezimal-Hexadezimal-ASCII-Tabelle	Seite 102
Anhang C: Der Farbspeicher	Seite 104
Anhang D: Softwareliste	Seite 105

Einige allgemeine Hinweise

Bei allen Programmlistings in diesem Buch sind folgende Zeichenumwandlungen zu beachten:

'S' entspricht dem '@'
'A' entspricht dem '['

Die meisten Basic-Programme in diesem Buch sind 'entpackt', d.h. nach jedem Befehl folgt ein Leerzeichen, in jeder Zeile steht nur ein Befehl usw. Es ist natürlich nicht nötig, alle diese Leerzeichen mit einzugeben, es erleichtert jedoch das Programmverständnis und die Fehlersuche.

Es empfiehlt sich, zunächst das 'Colour-Genie Buch I' zu lesen, da dort wesentlich einfachere Dinge erklärt und beschrieben werden. In diesem Buch werden sehr viele Probleme in Maschinensprache gelöst, dies erfordert natürlich gewisse Kenntnisse des Rechners, die Sie sich zunächst aneignen sollten.

Um sich in die Maschinensprache des Colour-Genie einzuarbeiten, empfehlen wir Ihnen:

- 'Programmierung des Z80' von Rodney Zaks, erschienen im Sybex-Verlag
- 'Das Colour-Genie ROM-Listing' von TCS

Ein Musikprogramm in Basic: Der 'BLUESMAKER'

Das folgende Basicprogramm erzeugt Melodien, die einem Blues nachempfunden sind.

Es sind dabei drei Grundakkorde vorgegeben, die in den DATA Zeilen 190 bis 210 stehen. Die erste Zahl entspricht der Oktave und die zweite der Tonhöhe, wie sie im PLAY Befehl benutzt werden. In Zeile 220 stehen die drei Töne für den Baß. Zeile 230 enthält codiert die Folge der Akkorde. In Zeile 240 bis 260 stehen die Töne, die bei der Melodie zum jeweiligen Akkord gespielt werden können.

Die Melodie wird während des Programmlaufs zufällig aus den zu dem Akkord passenden Noten ausgewählt. Mit der Eingabe des Tempos bestimmen Sie die Geschwindigkeit der Baßbegleitung.

Programmlisting:

```
100 DEFINT A-Z
110 CLS:COLOUR5
120 AS="COLOUR GENIE Bluesmaker"
130 FORL=1TO13
140 COLOUR(LAND15)+1
150 PRINTTAB(L)AS
160 NEXTL
170 PRINT:PRINT
180 INPUT"Tempo (1...6) ":TE
190 DATA 5,2,6,2,12,2,6,2
200 DATA 1,3,2,3,9,3,2,3
210 DATA 2,3,3,3,4,3,3,3
220 DATA 1,2,4,2,5,2
230 DATA1,1,1,1,2,2,2,2,1,1,1,1,3,2,1,1
240 DATA 1,4,9,4,3,4,10,4,5,4,12,4,1,5
250 DATA 1,4,9,4,4,4,10,4,5,4,12,4,1,5
260 DATA 5,3,12,3,7,3,8,4,2,4,4,4,5,4
270 DIM BT(3,4),B0(3,4)
280 DIM LT(3),LO(3)
290 DIM L(16)
300 DIM JT(3,7),JO(3,7)
310 FOR X=1TO3:FOR Y=1TO4
320 READ BT(X,Y),B0(X,Y)
330 NEXT Y,X
340 FOR X=1TO3
350 READ LT(X),LO(X)
360 NEXT X
370 FOR X=1TO16
380 READ L(X)
390 NEXT X
400 FOR X=1TO3:FOR Y=1TO7
410 READ JT(X,Y),JO(X,Y)
420 NEXT Y,X
```



```

430 C=1:T=1:B=0
440 GOSUB 470
450 GOSUB 630
460 GOTO 440
470 IF B<>1 THEN 580
480 PLAY (1,1,0,0)
490 PLAY (2,1,0,0)
500 D=D+1:IF D<>1 THEN RETURN
510 D=0:B=0
520 T=T+1
530 IF T<>5 THEN RETURN
540 T=1
550 P=P+1:IF P=17 THEN P=1
560 C=L(P)
570 RETURN
580 D=D+1
590 IF D=TEMPO THEN B=1:D=0
600 PLAY (1,LO(C),LT(C),8)
610 PLAY (2,BO(C,T),BT(C,T),8)
620 RETURN
630 NT=NT+AND(3)-2
640 IF NT<1 THEN NT=1
650 IF NT>7 THEN NT=7
660 PLAY (3,JO(C,NT),JT(C,NT),9)
670 RETURN

```

Enterprise - ein strategisches Weltraumspiel

(Für dieses Spiel muß Ihr Colour-Genie mit 32K RAM ausgerüstet sein. Das Programmlisting ist kommentiert, so daß wir Ihnen nur noch möglichst wenige Fehler beim Abtippen wünschen !)

SPIELERKLÄRUNG

1) Einleitung:

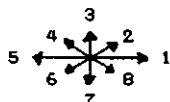
Enterprise ist ein Strategiespiel, bei dem es auf eine kluge Taktik ankommt. Man bewegt sich durch die Milchstraße auf der Suche nach Klingonen, die die friedlichen Planeten der Föderation mit Tod und Vernichtung überziehen. Sie sind nun der Kommandant des Sternschiffs ENTERPRISE, dem einzig verfügbaren Kampfraumer in dem Teil der Galaxis, wo die Klingonen zuletzt geortet worden sind. Unter Lebensgefahr müssen Sie die Klingonen bekämpfen und vernichten, um die Planeten vor dem endgültig Untergang zu bewahren. Zu Ihrer Unterstützung sind in der Galaxis einige Raumbasen verteilt, an die Sie bei Bedarf andocken können. Die Enterprise wird dann aufgetankt und evtl. repariert. Wenn Sie andocken wollen, müssen Sie nur in denselben Raumsektor steuern, in der die Basis steht.

2) Kommandos:

Ein Kommando wird vom Computer dann erwartet, wenn auf dem Bildschirm 'BEFEHL ?' erscheint. Wenn Sie einen falschen Befehl eingeben, wird eine Zusammenfassung der gültigen Befehle ausgegeben.

NAV -> Steuern

Mit diesem Kommando können Sie die Enterprise durch die Galaxis bewegen. Die Galaxis ist in Quadranten und Sektoren unterteilt. Wollen Sie Klingonen bekämpfen, so müssen sich diese in Ihrem Quadranten aufhalten. Um einen solchen Quadranten zu erreichen, geben Sie den Befehl NAV ein. Sie werden dann nach Richtung und Länge Ihres Kurses gefragt. Der Wert für die Richtung liegt zwischen 1 und 8 und bedeutet folgendes:



Die Eingabe von Richtungswerten erlaubt auch Kommazahlen damit eine genaue Steuerung möglich ist. Die Kurslänge wird durch den Warpfaktor festgelegt. Die Zahl vor dem Komma gibt dabei die Anzahl der zu durchfliegenden Quadranten und die Nachkommazahl die der Sektoren an. Aber Vorsicht ist geboten ! Sollte ein Stern auf Ihrer Flugbahn liegen, so werden die Triebwerke automatisch abgeschaltet und die Energie für den Flug geht verloren.

NAT -> Nahabtaster

Der Nahabtaster zeigt Ihnen ein Bild des Quadranten, in dem sich die ENTERPRISE aufhält, sowie einen verkürzten Status-report. Ein Quadrant ist in 8 mal 8 Sektoren unterteilt.

Die Symbole bedeuten:

Enterprise	E
Klingone	K
Raumbasis	B
Stern	*

FAT -> Fernabtaster

Der Fernabtaster zeigt den Inhalt des eigenen und der umliegenden acht Quadranten an. Der dreistelligen Zahl entnehmen Sie:

1. Ziffer - Anzahl der Klingonen
2. Ziffer - Anzahl der Raumstationen
3. Ziffer - Anzahl der Sterne

PHA -> Phaser

Mit dem Phaser kann man alle Klingonen im Quadranten mit der Energie beschießen, die man angibt. Ist die abgefeuerte Energiemenge zu knapp gewesen, so greifen die Klingonen ihrerseits die ENTERPRISE und können erheblichen Schaden anrichten (Status-Report beachten!).

TOR -> Torpedo

Sie feuern ein Torpedo ab, indem Sie wie beim Navigieren den Kurs angeben. Sollten nach dem Abschuß des Torpedos noch Klingonen vorhanden sein, so schießen diese ebenfalls auf die ENTERPRISE. Der Torpedokurs kann über COM, Option2 vom Bordcomputer errechnet werden.

SCH -> Schutzschirm

Dieser Befehl verändert die Energie der Schutzschirme. Wird die ENTERPRISE angegriffen, so sind diese Schirme das einzige Mittel einen Kampf überstehen. Ohne Energie auf den Schutzschirmen wäre Ihr Schiff beim geringsten Treffer vernichtet.

COM -> Computerabfrage

Nachdem Sie dieses Kommando eingegeben haben, wird die Frage nach einer Option gestellt. Diese Option muß eine Zahl zwischen 0 und 4 sein, die folgende Bedeutungen haben:

- 0: Zeigt alle bisher erfolgten Fernabtastungen.
(Code siehe oben)
- 1: Gibt einen Statusreport
- 2: Torpedokursberechnung (Richtung/Länge)
- 3: Rechnet Kurs und Geschwindigkeit zur Sternbasis in Ihrem Quadranten aus. Der Warpfaktor beträgt nur 1/10 des Geschwindigkeitswertes.
- 4: Berechnet den Kurs zwischen zwei Punkten.

Lassen Sie sich durch die Vielfalt der Kommandos nicht verwirren. Nach einiger Übung wird Ihnen jeder Befehl in Fleisch und Blut übergehen. Etwaige Unklarheiten werden sicherlich im Spiel geklärt werden.

Programmlisting:

```
10 ' SUPER STAR TREK
20 '
30 RANDOM
40 CLEAR 600
50 COLOUR 3
60 CLS
70 PRINT"Der Weltraum.... unendliche Weiten.... "
80 COLOUR 5
90 PRINT"Wir schreiben das Jahr 2163 alter
100 PRINT"Zeitrechnung. Die ENTERPRISE ist mit
110 PRINT"ihrer ueber 400 Mann starken Besatzung
120 PRINT"Unterwegs um neue Welten zu erforschen.
130 PRINT"neues Leben und neue Zivilisation.
140 PRINT
150 COLOUR 11
160 PRINT"Sie dringt dabei in Galaxien vor, die
```

```

170 PRINT"nie ein Mensch zuvor gesehen hat !"
180 PRINT"Ihre Aufgabe als Captain der Enterprise
190 PRINT"ist es, Ihr Schiff und die Besatzung
200 PRINT"vor den Angriffen der feindlichen
210 PRINT"Klingonen zu schuetzen
220 PRINT
230 COLOUR 16
240 PRINT"Die Foederation wuenscht Ihnen
250 PRINT"          viel Erfolg.
260 Z$="          "
270 DIM G(8, 8), C(9, 2), K(3, 3), N(3), Z(8, 8), O(8)
280 T=( RND(20)+20)*100
290 T0=T
300 T9=RND(10)+25
310 D0=0
320 E=3000
330 E0=E
340 P=10
350 P0=P
360 S8=200
370 S=0
380 E9=0
390 K9=0
400 X$="basis"
410 X0$=" 1st"
420 '  INIT ENTERPRISE POSITION
430 Q1=RND(8)
440 Q2=RND(8)
450 S1=RND(8)
460 S2=RND(8)
470 FOR I=1 TO 9
480 C(I, 1)=0
490 C(I, 2)=0
500 NEXT
510 C(3, 1)=-1
520 C(2, 1)=-1
530 C(4, 1)=-1
540 C(4, 2)=-1
550 C(5, 2)=-1
560 C(6, 2)=-1
570 C(1, 2)=1
580 C(2, 2)=1
590 C(6, 1)=1
600 C(7, 1)=1
610 C(8, 1)=1
620 C(9, 2)=1
630 C(8, 2)=1
640 FOR I=1 TO 8

```

```

650 D(I)=0
660 NEXT
670 A1$="NAVNATFATPHATORSCHZERCOM"
680 ' GALAXIS AUFBAUEN
690 ' K3=Klingonen B3=Basis S3=Stern
700 FOR I=1 TO 8
710 FOR J=1 TO 8
720 K3=0
730 Z(I, J)=0
740 R1=AND(0)
750 IF R1>.98 THEN K3=3: K9=K9+3: GOTO 780
760 IF R1>.95 THEN K3=2: K9=K9+2: GOTO 780
770 IF R1>.80 THEN K3=1: K9=K9+1
780 B3=0
790 IF AND(0)>.96 THEN B3=1: B9=B9+1
800 G(I, J)=K3*100+B3*10+AND(8)
810 NEXT
820 NEXT
830 IF K9>T9 THEN T9=K9+1
840 IF B9<>0 THEN 900
850 IF G(Q1, Q2)<200 THEN G(Q1, Q2)=G(Q1, Q2)+100: K9=K9+1
860 B9=1
870 G(Q1, Q2)=G(Q1, Q2)+10
880 Q1=AND(8)
890 Q2=AND(8)
900 K7=K9
910 IF B9<>1 THEN X$="basen": X0$=" sind"
920 COLOUR AND(16)
930 PRINT @ 930,"Druecke <RETURN>"
940 IF(PEEK(&HF840)AND 1)=0 THEN 920
950 CLS
960 COLOUR 16
970 PRINT"Das Oberkommando gibt Dir den Auftrag:"
980 PRINT
990 PRINT"Zerstoere":K9:"Klingonische Raumschiffe."
1000 PRINT"die in Foederationsgebiet eingefallen"
1010 PRINT"sind,bevor sie das Hauptquartier"
1020 PRINT"angreifen koennen."
1030 COLOUR 7
1040 PRINT
1050 PRINT"D Du hast bis STARDATE":T0+T9:"Zeit,also":T9
1060 PRINT"Einheiten"
1070 COLOUR 6
1080 PRINT
1090 PRINT"Es":X0$: B9:"Sternen":X$:" im eventuellen"
1100 PRINT"Kampfgebiet,um Dich zu versorgen"
1110 ' QUADRANT BERECHNEN

```

```

1120 Z4=Q1
1130 Z5=Q2
1140 K3=0
1150 B3=0
1160 S3=0
1170 G5=0
1180 D4=RND(.5)
1190 Z(Q1, Q2)=G(Q1, Q2)
1200 IF Q1<1 OR Q2<1 OR Q1>8 OR Q2>8 THEN 1470
1210 GOSUB 4940
1220 PRINT
1230 IF TO<>T THEN 1320
1240 COLOUR 16
1250 PRINT
1260 PRINT"Deine Mission beginnt im galaktischen Quadranten ";
1270 COLOUR 5
1280 PRINT G2$:
1290 COLOUR 16
1300 PRINT"".
1310 GOTO 1330
1320 PRINT"Du kommst nach ";G2$;"".
1330 PRINT
1340 K3=INT(G(Q1, Q2)*.01)
1350 B3=INT(G(Q1, Q2)*.1)-10*K3
1360 S3=G(Q1, Q2)-100*K3-10*B3
1370 IF K3=0 THEN 1430
1380 COLOUR 3
1390 PRINT"KAMPFGEBIET          CONDITION ROT"
1400 IF S>200 THEN 1430
1420 PRINT"SCHUTZSCHIRME GEFAEHRlich SCHWACH"
1430 FOR I=1 TO 3
1440 K(I, 1)=0
1450 K(I, 2)=0
1460 NEXT
1465 COLOUR 16
1470 FOR I=1 TO 3
1480 K(I, 3)=0
1490 NEXT
1500 Q$=Z$+Z$+Z$+Z$+Z$+Z$+Z$+LEFT$(Z$, 17)
1510 ' POSITIONEN IN QUADRANTEN FESTLEGEN
1520 A$="<E>"
1530 Z1=S1
1540 Z2=S2
1550 GOSUB 4870
1560 IF K3<1 THEN 1670
1570 FOR I=1 TO K3

```

```

1580 GOSUB 4780
1590 A$=" "+K+"
1600 Z1=R1
1610 Z2=R2
1620 GOSUB 4870
1630 K(I, 2)=R2
1640 K(I, 1)=R1
1650 K(I, 3)=S9*(.5+RND(0))
1660 NEXT
1670 IF B3<1 THEN 1750
1680 GOSUB 4780
1690 A$=" ">B<"
1700 Z1=R1
1710 B4=R1
1720 Z2=R2
1730 B5=R2
1740 GOSUB 4870
1750 FOR I=1 TO S3
1760 GOSUB 4780
1770 A$=" " * "
1780 Z1=R1
1790 Z2=R2
1800 GOSUB 4870
1810 NEXT
1820 GOSUB 5460
1830 IF S+E>10 THEN IF E>10 OR D(7)=0 THEN 1870
1840 PRINT
1850 PRINT"          SCHWERER FEHLER "
1860 GOTO 4640
1870 INPUT"Befehl ";A$
1880 FOR I=1 TO 9
1890 IF LEFT$(A$, 3)<>MID$(A1$, 3*I-2, 3)THEN 1910
1900 ON I GOTO 2030, 1820, 3190, 3410, 3790, 4480, 6530, 7280
1910 NEXT
1920 PRINT
1930 PRINT"FOLGENDE EINGABEN SIND MOEGlich"
1940 PRINT" NAV => BEWEGEN"
1950 PRINT" NAT => NAHABTASTER"
1960 PRINT" FAT => FERNABTASTER"
1970 PRINT" PHA => PHASER"
1980 PRINT" TOR => TORPEDO"
1990 PRINT" SCH => SCHUTZSCHIRME"
2000 PRINT" ZER => SCHADENSMELDUNG"
2010 PRINT" COM => COMPUTERANFRAGE"
2020 GOTO 1830
2030 ' FLUGHKONTROLLE
2040 INPUT"Richtung ";C1
2050 IF C1=9 THEN C1=1

```



```

2060 IF C1<1 OR C1>9 THEN PRINT: PRINT" Ltn. Sulu:Falscher Kurs.Sir !"
: GOTO 1830
2070 X$="8"
2080 IF D(1)<0 THEN X$="0.2"
2090 PRINT"Warp Faktor (0-":X$;")";
2100 INPUT W1
2110 IF D(1)<0 AND W1>.2 THEN 2160
2120 IF W1>0 AND W1<=8 THEN 2190
2130 IF W1=0 THEN 1830
2140 PRINT
2150 PRINT"Cheftechniker Scott: Warp";W1;" ? Das      schaffen unsere M
aschinen nicht !"
2160 PRINT"Warptriebwerke beschaedigt !"
2170 PRINT"      Maximale Beschleunigung:0.2 Warp"
2180 GOTO 1830
2190 N=INT(W1*8+.5)
2200 IF E-N>=0 THEN 2280
2210 PRINT
2220 PRINT"Maschinenraum:Fuer Warp";W1;" reicht die      Energie nicht au
s !"
2230 IF S<N-E OR D(7)<0 THEN 1830
2240 PRINT
2250 PRINT"Schirakontrolle:Schilde sind mit";S
2260 PRINT"Energieeinheiten geladen !"
2270 GOTO 1830
2280 * BEWEGEN & FEUERN DER KLINGONEN
2290 FOR I=1 TO K3
2300 IF K(I, 3)=0 THEN 2400
2310 A$="  "
2320 Z1=K(I, 1)
2330 Z2=K(I, 1)
2340 GOSUB 4870
2350 GOSUB 4780
2360 K(I, 1)=Z1
2370 K(I, 2)=Z2
2380 A$=" "+K+" "
2390 GOSUB 4870
2400 NEXT
2410 GOSUB 6330
2420 D1=0
2430 D6=W1
2440 IF W1>=1 THEN D6=1
2450 FOR I=1 TO 8
2460 IF D(I)>=0 THEN 2510
2470 D(I)=D(I)+D6
2480 IF D(I)>=.1 AND D(I)<0 THEN D(I)--.1: GOTO 2510
2490 IF D(I)<0 THEN 2510
2500 IF D(I)<>1 THEN D1=1: R1=I: GOSUB 6060: PRINT"Schadensmeldung: ";R

```

```

2510 NEXT
2520 IF AND(0)>.2 THEN 2640
2530 R1=AND(8)
2540 IF AND(0)>=.6 THEN 2600
2550 D(R1)=D(R1)-( AND(5))
2560 PRINT"Schadensmeldung:";
2570 GOSUB 6060
2580 PRINT G2$;" beschadigt"
2590 GOTO 2690
2600 D(R1)=D(R1)+AND(3)
2610 PRINT"Schadensmeldung:";
2620 GOSUB 6060
2630 PRINT" reparieren schreitet voran"
2640 ' ENTERPRISE WIRD BEWEGT
2650 A$=" "
2660 Z1=INT(S1)
2670 Z2=INT(S2)
2680 GOSUB 4870
2690 X1=C(C1, 1)+(C(C1+1, 1)-C(C1, 1))*(C1-INT(C1))
2700 X=S1
2710 Y=S2
2720 X2=C(C1, 2)+(C(C1+1, 2)-C(C1, 2))*(C1-INT(C1))
2730 Q4=Q1
2740 Q5=Q2
2750 FOR I=1 TO N
2760 S1=S1+X1
2770 S2=S2+X2
2780 IF S1<1 OR S1>=9 OR S2<1 OR S2>=9 THEN 3000
2790 S8=INT(S1)*24+INT(S2)*3-26
2800 IF MID$(Q$, S8, 2)=" " THEN 2850
2810 S1=INT(S1-X1)
2820 S2=INT(S2-X2)
2830 PRINT"Warptriebwerke schalteten sich wegen schlechter Navigati
on in Sektor ";S1;" ";S2;" ab"
2840 GOTO 2880
2850 NEXT
2860 S1=INT(S1)
2870 S2=INT(S2)
2880 A$="<E>"
2890 Z1=INT(S1)
2900 Z2=INT(S2)
2910 GOSUB 4870
2920 GOSUB 6240
2930 T8=1
2940 IF W1<1 THEN T8=.1*INT(W1*10)
2950 T=T+T8
2960 IF T>T0+T9 THEN 4640
2970 ' IST ENTERPRISE ANGEDOCKT ?
2980 GOTO 1820
2990 '

```

```

3000 X=8*Q1+X+N*X1
3010 Y=8*Q2+Y+N*X2
3020 Q1=INT(X/8)
3030 Q2=INT(Y/8)
3040 S1=INT(X-Q1*8)
3050 S2=INT(Y-Q2*8)
3060 IF S1=0 THEN Q1=Q1-1: S1=8
3070 IF S2=0 THEN Q2=Q2-1: S2=8
3080 X5=0
3090 IF Q1<1 THEN X5=1: Q1=1: S1=1
3100 IF Q1>8 THEN X5=1: Q1=8: S1=8
3110 IF Q2<1 THEN X5=1: Q2=1: S2=1
3120 IF Q2>8 THEN X5=1: Q2=8: S2=8
3130 IF X5<>0 THEN PRINT: PRINT"Ltn. Uhura: Es ist uns verboten,die
    Galaxis zu verlassen.Die Triebwerke    sollen abgestellt werden !":
    PRINT"Cheftechniker Scott:Triebwerke gestoppt"
3135 PRINT" Cpt. Chekov: Wir sind im Sektor";S1;",";S2;"des Quadranten
";
3140 IF T>T0+T9 THEN 4640
3150 IF 8*Q1+Q2=8*Q4+Q5 THEN 2880
3160 T=T+1
3170 GOSUB 6240
3180 GOTO 1120
3190 ' FERNRADAR
3200 '
3210 IF D(3)<0 THEN PRINT"Fernradar faellt aus": GOTO 1830
3220 PRINT"Fernradar Auswertung fuer
Quadrant";Q1;",";Q2
3230 O1$="-----"
3240 PRINT O1$
3250 FOR I=Q1-1 TO Q1+1
3260 N(1)=-1
3270 N(2)=-2
3280 N(3)=-3
3290 FOR J=Q2-1 TO Q2+1
3300 IF I>0 AND I<9 AND J>0 AND J<9 THEN N(J-Q2+2)=G(I, J): Z(I, J)=G(
I, J)
3310 NEXT
3320 FOR L=1 TO 3
3330 PRINT": ";
3340 IF N(L)<0 THEN PRINTSTRING$(3,195);" ": GOTO 3360
3350 PRINT RIGHT$(STR$(N(L)+1000), 3);" ":
3360 NEXT
3370 PRINT": "
3380 PRINT O1$
3390 NEXT
3400 GOTO 1830
3410 ' PHASER KONTROLLE
3420 '
3430 IF D(4)<0 THEN PRINT"Phaser sind inaktiv": GOTO 1830

```

```

3440 IF K3<0 THEN PRINT"IO Spock:Sensoren zeigen keine Klingonen in di
ese Quadranten": GOTO 1830
3450 IF D(8)<0 THEN PRINT"Fehler im Feuerleitcomputer"
3460 PRINT"Phaser im Ziel"
3470 PRINT"Verfuegbare Energie: ";E;" Einheiten"
3480 INPUT"Schussenergie ";X
3490 IF X<=0 THEN 1830
3500 IF E-X<0 THEN 3470
3510 E=E-X
3520 IF D(7)<0 THEN X=X*AND(0)
3530 H1=INT(X/K3)
3540 FOR I=1 TO 3
3550 IF K(I, 3)<=0 THEN 3760
3560 H=INT((H1/SQR((K(I, 1)-S1)A2+(K(I, 2)-S2)A2)))*( AND(0)+2))
3570 IF H>.15*K(I, 3) THEN 3600
3580 PRINT"Sensoren zeigen keinen Treffer"
3590 GOTO 3760
3600 K(I, 3)=K(I, 3)-H
3610 PRINT H;" Einheiten trafen Klingonen in":PRINT"Sektor";K(I, 1);",
";K(I, 2)
3620 IF K(I, 3)<=0 THEN PRINT" *** KLINGONE ZERSTOERT ***": GOTO 36
60
3630 PRINT" (Sensoren zeigen";K(I, 3)
3640 PRINT" verbleibende Einheiten)"
3650 GOTO 3760
3660 K3=K3-1
3670 K9=K9-1
3680 Z1=K(I, 1)
3690 Z2=K(I, 2)
3700 A$=" "
3710 GOSUB 4870
3720 K(I, 3)=0
3730 G(Q1, Q2)=G(Q1, Q2)-100
3740 Z(Q1, Q2)=G(Q1, Q2)
3750 IF K9<=0 THEN 7830
3760 NEXT
3770 GOSUB 6330
3780 GOTO 1830
3790 ' PHOTONENTORPEDO KONTROLLE
3800 '
3810 IF P<=0 THEN PRINT"Keine Torpedos mehr verfuegbar": GOTO 1830
3820 IF D(5)<0 THEN PRINT"Agschusseinrichtung ausser Funktion": GOTO 1
830
3830 INPUT"Torpedokurs (1-9) ";C1
3840 IF C1=9 THEN C1=1
3850 IF C1<1 OR C1>9 THEN PRINT"Chekov:Falsche Kursdaten.Sir!": GOTO 1
830
3860 X1=C(C1, 1)+(C(C1+1, 1)-C(C1, 1))*(C1-INT(C1))

```

```

3870 E=E-2
3880 P=P-1
3890 X2=C(C1, 2)+(C(C1+1, 2)-C(C1, 2))*(C1-INT(C1))
3900 X=S1
3910 Y=S2
3920 PRINT"Torpedo fliegt"
3930 X=X+X1
3940 Y=Y+X2
3950 X3=INT(X+.5)
3960 Y3=INT(Y+.5)
3970 IF X3<1 OR X3>8 OR Y3<1 OR Y3>8 THEN 4450
3980 PRINT"          ";X3;" ";Y3
3990 A$="  "
4000 Z1=X
4010 Z2=Y
4020 GOSUB 5390
4030 IF Z3<>0 THEN 3930
4040 A$=" "+K+" "
4050 Z1=X
4060 Z2=Y
4070 GOSUB 5390
4080 IF Z3=0 THEN 4190
4090 PRINT"*** Klingone zerstört ***"
4100 K3=K3-1
4110 K9=K9-1
4120 IF K9<=0 THEN 7830
4130 FOR I=1 TO 3
4140 IF X3=K(I, 1)AND Y3=K(I, 2)THEN 4170
4150 NEXT
4160 I=3
4170 K(I, 3)=0
4180 GOTO 4370
4190 A$=" * "
4200 Z1=X
4210 Z2=Y
4220 GOSUB 5390
4230 IF Z3<>0 THEN PRINT"Stern bei";X3;" ";Y3;" absorbierte Torpedo";
GOSUB 6330: GOTO 1830
4240 A$=">B<"
4250 Z1=X
4260 Z2=Y
4270 GOSUB 5390
4280 IF Z3=0 THEN 3820
4290 PRINT"*** Sternenbasis zerstört ***"
4300 B3=B3-1
4310 B9=B9-1
4320 IF B9>0 OR K9>T-T0-T9 THEN 4350
4330 PRINT"Das wars.Kapitaen!!!"
4340 PRINT"Du bist hiermit Deines Kommandos entzogenund zu 99 Sternenta
gen Strafarbeit auf Cygnus 12 verdonnert !!!"
4350 PRINT"Das Flottenkommando sammelt Anklage-      punkte gegen Dich"

```

```

4360 D0=0
4370 Z1=X
4380 Z2=Y
4390 A$=""
4400 GOSUB 4870
4410 G(Q1, Q2)=K3*100+B3*10+S3
4420 Z(Q1, Q2)=G(Q1, Q2)
4430 GOSUB 6330
4440 GOTO 1830
4450 PRINT"Torpedo ging am Ziel vorbei"
4460 GOSUB 6330
4470 GOTO 1830
4480 ' SCHIRMKONTROLLE
4490 IF D(7)<0 THEN PRINT"Schirmkontrolle ausgefallen": GOTO 1830
4500 PRINT"Verfuegbare Energie:":E+S
4510 INPUT"Wieviel Energie in die Schirme ";X
4520 IF X<0 OR S=X THEN PRINT"<Schirme unveraendert>": GOTO 1830
4530 IF X>E+S THEN PRINT"Schirmkontrolle:Wir sind nicht die Schatzkammer der Foederation": PRINT" <Schirme unveraendert>": GOTO 1830
4540 E=E+S-X
4550 S=X
4560 PRINT"Deflektorraum:Schilde jetzt auf"
4570 PRINT INT(S):" Einheiten"
4580 GOTO 1830
4590 ' ENDE DES SPIELS
4600 CLS
4610 IF T>T0+T9 THEN PRINT"Deine Zeit ist ueberschritten,das Haupt-quartier ist verloren und mit ihm die Galaxis,Du Niete '!!!!': END
4620 PRINT"Heute ist der Sternentag":T
4630 GOTO 4680
4640 PRINT
4650 PRINT" DIE ENTERPRISE IST ZERSTOERT"
4660 PRINT" DIE FOEDERATION IST VERLOREN"
4670 GOTO 4620
4680 IF K9>1 THEN PRINT"Schade,es waren noch":K9;" Klingonische": PRINT"Schlachtschiffe uebrig" ELSE PRINT"So ein Mist,haettest Du das letzte": PRINT"Schiff doch noch erwischt,Pech!"
4690 PRINT"Das ist das Ende Deiner Mission !"
4700 PRINT
4710 PRINT
4720 IF B9=0 THEN END
4730 PRINT"Die Foederation braucht einen neuen Kommandanten,also Freiwillige vor !!!!!"
4740 PRINT
4750 INPUT"Tritt heran und tippe 'AYE' ":A$
4760 IF A$="AYE" THEN RUN
4770 END
4780 ' LEEREN PLATZ IM QUADRANTEN FINDEN
4790 R1=RND(8)

```

```

4800 R2=RND(8)
4810 A$="  "
4820 Z1=R1
4830 Z2=R2
4840 GOSUB 5390
4850 IF Z3=0 THEN 4790
4860 RETURN
4870 '
4880 S8=INT(Z2-.5)*3+INT(Z1-.5)*24+1
4890 IF LEN(A$)>3 THEN PRINT"FEHLER": STOP
4900 IF S8=1 THEN Q$=A$+RIGHT$(Q$, 189): RETURN
4910 IF S8=190 THEN Q$=LEFT$(Q$, 189)+A$: RETURN
4920 Q$=LEFT$(Q$, S8-1)+A$+RIGHT$(Q$, 190-S8)
4930 RETURN
4940 ' NAME DES GEBIETS BESTIMMEN
4950 IF Z5<=4 THEN ON Z4 GOTO 4970, 4990, 5010, 5030, 5050, 5070, 5090
, 5110
4960 GOTO 5130
4970 G2$="Antares"
4980 GOTO 5120
4990 G2$="Rigel"
5000 GOTO 5120
5010 G2$="Procyon"
5020 GOTO 5120
5030 G2$="Wega"
5040 GOTO 5120
5050 G2$="Canopus"
5060 GOTO 5120
5070 G2$="Altair"
5080 GOTO 5120
5090 G2$="Sagittarius"
5100 GOTO 5120
5110 G2$="Pollux"
5120 GOTO 5290
5130 ON Z4 GOTO 5140, 5160, 5180, 5200, 5220, 5240, 5260, 5280
5140 G2$="Sirius"
5150 GOTO 5290
5160 G2$="Deneb"
5170 GOTO 5290
5180 G2$="Capella"
5190 GOTO 5290
5200 G2$="Beteigeuze"
5210 GOTO 5290
5220 G2$="Aldebaran"
5230 GOTO 5290
5240 G2$="Regulus"
5250 GOTO 5290
5260 G2$="Arcturus"

```

```

5270 GOTO 5290
5280 G2$="Spica"
5290 IF G5<>1 THEN ON Z5 GOTO 5310, 5330, 5350, 5370, 5310, 5330, 5350
, 5370
5300 RETURN
5310 G2$=G2$+" I"
5320 RETURN
5330 G2$=G2$+" II"
5340 RETURN
5350 G2$=G2$+" III"
5360 RETURN
5370 G2$=G2$+" IV"
5380 RETURN
5390 Z1=INT(Z1+.5)
5400 Z2=INT(Z2+.5)
5410 S8=(Z2-1)*3+(Z1-1)*24+1
5420 Z3=0
5430 IF MID$(Q$, S8, 3)<>A$ THEN RETURN
5440 Z3=1
5450 RETURN
5460 ' ANFANGSBILO UND NAHABTASTER
5470 '
5480 FOR I=S1-1 TO S1+1
5490 FOR J=S2-1 TO S2+1
5500 IF INT(I+.5)<1 OR INT(I+.5)>8 OR INT(J+.5)<1 OR INT(J+.5)>8 THEN
5560
5510 A$=" ">8<"
5520 Z1=I
5530 Z2=J
5540 GOSUB 5390
5550 IF Z3=1 THEN 5600
5560 NEXT
5570 NEXT
5580 D0=0
5590 GOTO 5670
5600 D0=1
5610 C$="DOCK"
5620 E=E0
5630 P=P0
5640 PRINT"Schirme werden fuers andocken abgebaut"
5650 S=0
5660 GOTO 5700
5670 IF K3>0 THEN C$="*ROT*": GOTO 5700
5680 C$="*GRUEN*"
5690 IF E<E0*.1 THEN C$="*GELB*"
5700 IF D(2)>=0 THEN 5750
5710 PRINT
5720 PRINT"*** keine Nahabtastung moeglich ***"

```



```

5730 PRINT
5740 RETURN
5750 O1$=CHR$(223)+STRING$(16,211)+CHR$(227)
5760 O2$=CHR$(227)+STRING$(16,218)+CHR$(223)
5770 PRINT O1$
5780 FOR I=1 TO 8
5790 PRINTCHR$(207);
5800 FOR J=(I-1)*24+1 TO(I-1)*24+22 STEP 3
5810 PRINT" ";MID$(Q$, J+1, 1);
5820 NEXT
5830 ON I GOTO 5840, 5860, 5920, 5940, 5960, 5980, 6000, 6020
5840 PRINTCHR$(219);" Sternentag";INT(T*10)*.1
5850 GOTO 6030
5860 PRINTCHR$(219);" Condition ";
5870 IF C$="*ROT*" THEN COLOUR 3:GOTO5890
5880 COLOUR 2
5890 PRINT C$
5900 COLOUR 16
5910 GOTO 6030
5920 PRINTCHR$(219);" Quadrant ";Q1;",";Q2
5930 GOTO 6030
5940 PRINTCHR$(219);" Sektor ";S1;",";S2
5950 GOTO 6030
5960 PRINTCHR$(219);" Torpedos ";INT(P)
5970 GOTO 6030
5980 PRINTCHR$(219);" Energie ";INT(E+S)
5990 GOTO 6030
6000 PRINTCHR$(219);" Schirae ";INT(S)
6010 GOTO 6030
6020 PRINTCHR$(219);" Klingonen ";INT(K9)
6030 NEXT
6040 PRINT O2$
6050 RETURN
6060 ' WAS WURDE ZERSTOERT
6070 ON R1 GOTO 6080, 6100, 6120, 6140, 6160, 6180, 6200, 6220
6080 G2$="Warp Triebwerk"
6090 RETURN
6100 G2$="Nahabtaster"
6110 RETURN
6120 G2$="Fernabtaster"
6130 RETURN
6140 G2$="Phaserkontrolle"
6150 RETURN
6160 G2$="Photonenroehren"
6170 RETURN
6180 G2$="Schadenskontrolle"
6190 RETURN
6200 G2$="Schirmkontrolle"

```

```

6210 RETURN
6220 G2$="Computer"
6230 RETURN
6240 ' ENERGIEVERBRAUCH
6250 '
6260 E=E-N-10
6270 IF E>=0 THEN RETURN
6280 PRINT"Zum beenden des Maneuvers muss Schirm- energie abgezogen w
erden !"
6290 S=S+E
6300 E=0
6310 IF S<=0 THEN S=0
6320 RETURN
6330 ' KLINGONEN FEUERN
6340 IF K3<=0 THEN RETURN
6350 IF D0<>0 THEN PRINT"Schilder der Sternbasis schuetzten Enter
prise": RETURN
6360 FOR I=1 TO 3
6370 IF K(I, 3)<=0 THEN 6510
6380 H=INT((K(I, 3)/SQR((K(I, 1)-S1)2+(K(I, 2)-S2)2))*(2+RND(0)))
6390 S=S-H
6400 K(I, 3)=K(I, 3)/(3+RND(0))
6410 PRINT H:" Einheiten trafen Enterprise vom "
6420 PRINT"Sektor":K(I, 1);",":K(I, 2)
6430 IF S<=0 THEN 4590
6440 PRINT"<Schilder auf":S:" Einheiten gesunken)"
6450 IF H<20 THEN 6510
6460 IF RND(0)>.6 OR H/S<=.02 THEN 6510
6470 R1=RND(8)
6480 D(R1)=D(R1)-H/S-.5*RND(0)
6490 GOSUB 6060
6500 PRINT"Schadensmeldung: ":G2$:" zerstort"
6510 NEXT
6520 RETURN
6530 ' SCHADENSMELDUNG
6540 IF D(6)>=0 THEN 6730
6550 PRINT"Schadensmeldung ausser Funktion"
6560 IF D0=0 THEN 1830
6570 D3=0
6580 FOR I=1 TO 8
6590 IF D(I)<0 THEN D3=D3+.1
6600 NEXT
6610 IF D3=0 THEN 1830
6620 PRINT
6630 D3=D3+D4
6640 IF D3>=1 THEN D3=.9
6650 PRINT"Techniker stehen bereit,um die Schaeden zu beheben"
6660 PRINT"Vorgesehene Zeit:":.01*INT(100*D3):" Sternentage"

```

```

6670 INPUT"Erlaubst Du die Reparatur (J/N) ";A$
6680 IF A$<>"J" THEN 1830
6690 FOR I=1 TO 8
6700 IF D(I)<0 THEN D(I)=0
6710 NEXT
6720 T=T+D3+.1
6730 PRINT
6740 PRINT
6750 PRINT"    Objekt                Reparaturstadium"
6760 FOR R1=1 TO 8
6770 GOSUB 6060
6780 PRINT G2$: LEFT$(Z$, 30-LEN(G2$)); INT(D(R1)*100)*.01
6790 NEXT R1
6800 PRINT
6810 IF D0<>0 THEN 6570
6820 GOTO 1830
6830 '    PHOTONENTORPDATEN
6840 '
6850 IF K3<=0 THEN PRINT"Keine Klingonen in Reichweite "; GOTO 1830
6860 IF K3=1 THEN X$="zum" ELSE X$="zu den"
6870 PRINT"Von der Enterprise ";X$;" Klingonen"
6880 H8=0
6890 FOR I=1 TO 3
6900 IF K(I, 3)<=0 THEN 7220
6910 W1=K(I, 1)
6920 X=K(I, 2)
6930 C1=S1
6940 A=S2
6950 GOTO 7010
6960 PRINT"Entfernungsbestimmung:"
6970 PRINT"D Du bist in Quadrant";Q1;" ";Q2:PRINT"    Sektoren";S
1;" ";S2:PRINT" Gebe ein
6980 '
6990 INPUT"Anfangskordinate (X,Y) ";C1, A
7000 INPUT"    Endkordinate (X,Y) ";W1, X
7010 X=X-A
7020 A=C1-W1
7030 IF X<0 THEN 7130
7040 IF A<0 THEN 7150
7050 IF X>0 THEN 7070
7060 IF A=0 THEN C1=5: GOTO 7080
7070 C1=1
7080 IF ABS(A)<=ABS(X) THEN 7110
7090 PRINT"Richtung:";C1+((( ABS(A)-ABS(X))+ABS(A))/ABS(A))
7100 GOTO 7200
7110 PRINT"Richtung:";C1+( ABS(A)/ABS(X))
7120 GOTO 7200
7130 IF A>0 THEN C1=3: GOTO 7160
7140 IF X>0 THEN C1=5: GOTO 7080

```

```

7150 C1=7
7160 IF ABS(A)>=ABS(X)THEN 7190
7170 PRINT"Richtung:";C1+((( ABS(X)-ABS(A))+ABS(X))/ABS(X))
7180 GOTO 7200
7190 PRINT"Richtung:";C1+( ABS(X)/ABS(A))
7200 PRINT"Distanz :";SQR(X2+A2)
7210 IF H8=1 THEN 1830
7220 NEXT
7230 GOTO 1830
7240 ' KURS ZUR BASIS
7250 IF B3(>)0 THEN PRINT"Von der Enterprise zur Basis:"; W1=B4; X=B5:
GOTO 6930
7260 PRINT"Mr.Spock:Die Sensoren zeigen keine Basis"
7270 GOTO 1830
7280 ' BUECHEREICOMPUTER
7290 IF D(8)<0 THEN PRINT"Computer inaktiv !": GOTO 1830
7300 INPUT"Computer ist bereit und wartet auf Eingabe ":A
7310 IF A<0 THEN 1830
7320 PRINT
7330 H8=1
7340 ON A+1 GOTO 7450, 7730, 6830, 7240, 6960
7350 PRINT
7360 PRINT" Abrufbare Funktionen : "
7370 PRINT" 0-Karte aller Abtastungen"
7380 PRINT" 1-Status Report"
7390 PRINT" 2-Photonentorpedodaten"
7400 PRINT" 3-Entfernung zur Basis"
7410 PRINT" 4-Entfernungsberechnung"
7430 PRINT
7440 GOTO 7300
7450 PRINT
7460 PRINT
7470 PRINT"Computeraufzeichnung der Galaxis ":Q1;"":Q2
7480 PRINT
7490 PRINT" 1 2 3 4 5 6 7 8"
7500 O1$=" "+STRING$(31,217)
7510 PRINT O1$
7520 FOR I=1 TO 8
7530 PRINT I;
7540 IF H8=0 THEN 7590
7550 FOR J=1 TO 8
7560 PRINT" F";
7570 IF Z(I, J)=0 THEN PRINTSTRING$(3,127):: GOTO 7590
7580 PRINT RIGHT$(STR$(Z(I, J)+1000), 3);
7590 NEXT
7600 PRINTCHR$(225);
7610 GOTO 7690
7620 Z4=1
7630 Z5=1
7640 J0=INT(15-.5*LEN(G2$))
7650 PRINT TAB(J0); G2$;

```

```

7660 Z5=5
7670 J0=INT(39-.5*LEN(G2$))
7680 PRINT TAB(J0); G2$;
7690 PRINT
7700 PRINT O1$
7710 NEXT
7720 GOTO 1830
7730 ' STATUS REPORT
7740 PRINT
7750 PRINT" Status Report:"
7760 IF K9>1 THEN X$="en" ELSE X$=""
7770 PRINT"Nach";K9;" Klingon";X$;" uebrig"
7780 PRINT"Die Mission muss in";.1*INT((T9+T0-T)*10);" Sternentagen"
7790 PRINT"beendet werden"
7800 IF B9>1 OR B9=0 THEN X$="basen" ELSE X$="basis"
7810 PRINT"Es gibt";B9;" Sternen";X$
7820 GOTO 6530
7830 ' GEWONNEN
7840 PRINT
7850 PRINT
7860 PRINT
7870 PRINT" DU HAST ALLE KLINGONEN VERNICHTET"
7880 PRINT"Dir gebuehrt der Dank der gesamten Milchstrasse!"
7890 PRINT"Die Foederation ernennt Dich zum"
7900 PRINT" ADMIRAL"
7910 PRINT"und gibt Dir das Kommando ueber die";RND(10);"Raumflotte.da
mit Du auch weiterhin Klin-gonen jagst !"
7920 IF INKEY$="" THEN 7920

```

Run to Morlock's Tower

Auf den folgenden Seiten finden Sie das Listing eines Abenteuerspiels, bei dem Sie durch Verhandeln, Kämpfen, Forschen u.s.w. versuchen sollen, das Endziel (Eroberung des 'Morlock-Towers') zu erreichen.

Im Programmlisting finden Sie ab Programmzeile 7150 die Spielanleitung. Beachten Sie, daß das eigentliche Spiel in Englisch abläuft - die wichtigen Begriffe werden größtenteils in der Anleitung erklärt, im schlimmsten Fall müssen Sie halt mal ein Wörterbuch zur Hand nehmen.

Für dieses Programm muß Ihr Colour-Genie mit 32K RAM ausgerüstet sein.

Programmlisting:

```
10 COLOUR16
20 GOSUB 8160
30 CLS
40 PRINT @ 400."BRAUCHEN SIE EINE ANLEITUNG (J/N)?" :CHR$(14);
50 AS=INKEY$
60 IFAS="N" THEN 80
70 IFAS="J" THEN GOTO 7150ELSE GOTO 50
80 CLEAR 132:PRINTCHR$(15)
90 DIM D$(9), O$(17), D(32, 4), O(17, 6)
100 CLS
110 GOSUB 150
120 GOSUB 8800
130 GOSUB 7040
140 GOTO 220
150 DATA 7, 255 , 8, 16, 9, 16, 10, 16, 13, 16
160 RESTORE
170 FOR I=1 TO 5
180 READ XX, YY
190 SOUND XX,YY
200 NEXT
210 RETURN
220 IF INKEY$ ="" THEN GOTO 220
230 CLS
240 C=17408
250 O(17, 1)=RND(155)+100
260 SP=RND(6)
270 ST=RND(10)
280 CH=RND(6)
290 M=10
300 MC=6
310 PL=880
320 L=40
330 BQ=223
340 S1=223
350 D$="want to"
360 W$="which item?"
370 E$="don't have any!"
380 H$="how many?"
390 M$="don't have that many!"
400 Y$="you're wasting my time!"
```

```

410 RS="they attacked anyway!"
420 DATA 67 CASTLE, 82 RUIN, 99 CAVE, 84 TOWER, 107 CITY, 80 PILGRIMS, 87 WIZ
ARDS, 79 OGRES, 83 SOLDIERS
430 DATA SIEGE TOWERS, SMALL BOAT, GRAPPLING HOOKS, OIL LAMPS, AXES, SHOVELS
, ROPE, LEATHER JERKINS, CHAIN MAIL, PLATE ARMOR, SHIELDS, BROAD SWORDS, LONG
BOWS, CROSSBOWS, RATIONS, LARGE KEY, GOLD COINS
440 DATA 1500, 2500, 150, 255, 15, 30, 5, 20, 10, 30, 10, 20, 15, 25,
15, 40, 2, 25, 60, 4, 50, 100, 8, 3, 20, 2, 10, 50, 5, 11, 50, 5, 18,
40, 4, 4, 15, 1, 5, 25, 1,
450 FOR I=1 TO 9
460 READ D$(I)
470 NEXT
480 FOR I=1 TO 17
490 READ O$(I)
500 NEXT
510 FOR I=1 TO 17
520 FOR J=2 TO 4
530 READ O(I, J)
540 NEXT J, I
550 DATA 67, 84, 99, 82, 67, 84, 99, 82, 67, 84, 99, 82, 67, 84, 99, 8
2, 67, 94, 99, 82, 67, 94, 99, 82, 80, 80, 87, 79, 83, 83
560 CLS
570 PRINT # 216, "please wait"
580 COLOUR 6
590 PRINT # 256, STRING$(11, 211)
600 COLOUR 16
610 D(1, 1)=107
620 D(2, 1)=107
630 D(1, 2)=RND(6)
640 D(2, 2)=RND(6)+8
650 D(1, 3)=RND(30)+1
660 D(2, 3)=RND(30)+1
670 FOR I=3 TO 32
680 D(I, 4)=1
690 READ D(I, 1)
700 D(I, 2)=RND(14)
710 D(I, 3)=RND(30)+1
720 IF D(I, 2)=8 THEN 700
730 FOR J=1 TO I
740 IF D(I, 2)=D(J, 2) THEN IF D(I, 3)=D(J, 3) THEN IF J<>I THEN 700
750 NEXT J, I
760 FOR I=8 TO 14
770 IF RND(10)>5 THEN O(I, 1)=RND(8)
780 NEXT
790 O(15, 1)=RND(25)+15
800 A=RND(7)
810 B=RND(32)
820 SL=RND(384)+17500
830 GOSUB 6610
840 N=C+B+A*L
850 F1=1
860 GOSUB 2150
870 GOSUB 3580
880 GOSUB 3360
890 GOSUB 150
900 EG=EG+1
910 IF DF THEN 1030
920 Q=1
930 F1=1
940 GOSUB 2150
950 GOSUB 3580

```

```

960 IF TW<=M*100 THEN 1030
970 GOSUB 7020
980 PRINT $ PL,"carring to much!";
990 GOSUB 6930
1000 GOSUB 2090
1010 GOSUB 6610
1020 GOTO 880
1030 GOSUB 7020
1040 COLOUR 2
1050 PRINT $ PL,"what now SIRE?";
1060 COLOUR 16
1070 A$="IUNSEW"
1080 GOSUB 1940
1090 IF Q=0 THEN 1080
1100 ON Q GOSUB 2090, 2660
1110 IF Q=1 OR Q=2 THEN GOSUB 6610
1120 IF Q=5 AND B<40 THEN B=B+1
1130 IF Q=4 AND A<15 THEN A=A+1
1140 IF Q=6 AND B>0 THEN B=B-1
1150 IF Q=3 AND A>0 THEN A=A-1
1160 IF A<>8 OR O(2, 1)>0 THEN 1230
1170 GOSUB 7020
1180 PRINT $ PL,"YOUR MEN CAN'T SWIM!";
1190 GOSUB 6930
1200 A=A-1
1210 IF Q=3 THEN A=A+2
1220 GOTO 880
1230 F=F+1
1240 P=O(15, 1)-M/2
1250 IF F<10 THEN 1300
1260 F=0
1270 O(15, 1)=P
1280 IF P<M THEN GOSUB 7020: PRINT $ PL,"FOOD LOW!";: GOSUB 6930:IF O(
15, 1)<0 THEN O(15, 1)=0
1290 IF P<M/2 THEN P=AND(M/3): GOSUB 7020: PRINT $ PL, P"men died!";:
M=M-P: GOSUB 6930
1300 MC=MC+1
1310 FOR I=1 TO 32
1320 IF A=D(I, 2) THEN IF B=D(I, 3) THEN POKE N, S1: GOTO 3640
1330 NEXT
1340 POKE N, S1
1350 N=C+A*L+B
1360 IF PEEK(N)<>131 THEN S1=PEEK(N)
1370 DF=0
1380 SWF=0
1390 MF=0
1400 GOSUB 7020
1410 PRINT $ PL,"you are in the ";
1420 B$="wilderness"
1430 IF A=8 THEN B$="river"
1440 IF B>18 THEN IF A<8 THEN B$="mountains": MF=1: F=F+.5
1450 IF S1=130 THEN B$="swamp": SWF=1
1460 IF S1=135 THEN PRINT"shadow of": GOSUB 6960: PRINT $ PL,"the dre
aded morlock's tower!": GOSUB 6960: GOTO 3390
1470 GOSUB 150
1480 '
1490 FOR J=0 TO 95
1500 GOSUB 150
1510 '
1520 FOR J=0 TO 140
1530 GOSUB 150

```



```

+ 1540
1550 PRINT B$;
1560 POKE N, 131
1570 GOSUB 6770
1580 BT=0
1590 FOR X=1 TO 32
1600 IF ABS(A-D(X, 2))>2 THEN 1710
1610 IF ABS(B-D(X, 3))>2 THEN 1710
1620 IF D(X, 4)=1 THEN D(X, 4)=0
1630 FOR Y=1 TO 9
1640 IF VAL(D$(Y))<>D(X, 1) THEN 1700
1650 IF Y>5 THEN BT=Y
1660 GOSUB 7020
1670 PRINT $ PL, MID$(D$(Y), 4); " near";
1680 GOSUB 6930
1690 GOTO 1710
1700 NEXT Y
1710 NEXT X
1720 IF A>12 THEN IF B>22 THEN GOSUB 7020: PRINT $ PL, "CITADEL near";:
GOSUB 6930
1730 IF SWF THEN IF AND(100)<50 THEN GOSUB 7020: PRINT $ PL, "2 men are
stuck in quicksand!";: M=M-2: DF=1: GOSUB 6930: GOTO 1750
1740 IF MF AND S1<>32 THEN IF AND(100)<50 THEN GOSUB 7020: PRINT $ PL,
"2 men fell into a deep ravine!";: M=M-2: DF=1: GOSUB 6930
1750 IF BT THEN 880
1760 BC=BC+1
1770 IF BC<3 THEN 880
1780 BC=0
1790 FOR J=23 TO 32
1800 IF D(J, 4)=0 THEN D(J, 4)=1: POKE C+D(J, 3)+D(J, 2)*L, BQ
1810 X=AND(15)
1820 Y=AND(40)
1830 IF X=8 THEN 1810
1840 FOR ZZ=1 TO 32
1850 IF X=D(ZZ, 2) THEN IF Y=D(ZZ, 3) THEN IF J<>ZZ THEN 1810
1860 NEXT ZZ
1870 D(J, 2)=X
1880 D(J, 3)=Y
1890 NEXT J
+ 1900 IF EX>0 THEN IF AND(100)>80 THEN J=AND(6)+22: D(J, 2)=A: D(J, 3)=
B: GOTO 1300
1910 N=C+B+A*L
1920 POKE N, 131
+ 1930 GOTO 880
1940 B$=INKEY$
1950 Q=0
1960 IF B$="" THEN 1940
1970 FOR I=1 TO LEN(A$)
1980 IF B$=MID$(A$, I, 1) THEN Q=I: RETURN
1990 NEXT
2000 RETURN
2010 A$=""
2020 Q=0
2030 GOSUB 7020
2040 B$=INKEY$
2050 IF B$<>"" THEN IF ASC(B$)=13 THEN Q=VAL(A$): RETURN
2060 IF B$<>"" THEN IF ASC(B$)=8 THEN 2010
2070 IF B$<>"" THEN A$=A$+B$: PRINT $ PL, A$;
2080 GOTO 2040
2090 CLS:COLOUR 2
2100 PRINT TAB(11)"INVENTORY"

```

```

2110 PRINT
2120 PRINT"ITEM          QUANT.  WT.  TOTAL"
2130 F1=0
2140 ZL=0
2150 WV=0
2160 TW=0
2170 FOR I=1 TO 17
2180 IF O(I, 1)=0 THEN 2260
2190 J=O(I, 1)
2200 M=O(I, 2)
2210 TW=TW+J*M
2220 IF F1=0 THEN GOSUB 2550: ZL=ZL+1:IF ZL=7 THEN GOSUB 6530: ZL=0
2230 X=O(I, 4)
2240 IF J>M THEN J=M
2250 WV=WV+J*X
2260 NEXT
2270 IF F1 THEN RETURN
2280 PRINT
2290 PRINT
2300 IF Q=3 THEN RETURN
2310 COLOUR 2:PRINT"TOTAL WEIGHT" TW" LBS."
2320 PRINT"your" M" men can carry" M*100
2330 IF Z=5 THEN 6080
2340 PRINT $ PL. D$ " drop anything?";
2350 FOR W=1 TO 30
2360 NEXT
2370 A$=INKEY$
2380 IF A$="" THEN 2370
2390 IF A$("<")"Y" THEN Q=1: RETURN
2400 GOSUB 7020
2410 PRINT $ PL. W$;
2420 GOSUB 6960
2430 GOSUB 2010
2440 IF Q>17 THEN 2400
2450 I=Q
2460 QT=O(I, 1)
2470 IF QT=0 THEN GOSUB 7020: PRINT $ PL. E$:: GOSUB 6960: GOTO 2340
2480 GOSUB 7020
2490 PRINT $ PL. H$;
2500 GOSUB 6960
2510 GOSUB 2010
2520 IF Q>QT THEN PRINT $ PL. M$:: GOSUB 6960: GOTO 2340
2530 O(I, 1)=QT-Q
2540 GOTO 2090
2550 COLOUR 5:PRINT MID$(STR$(I), 2) " ";
2560 IF I<10 THEN PRINT " ";
2570 PRINT O$(I): TAB(18)J: TAB(23)MID$(STR$(K), 2): TAB(27)MID$(STR$(
K*J), 2)
2580 RETURN
2590 Z=0
2600 BC=3
2610 BT=0
2620 B=B-1
2630 IF AND(10)>5 THEN B=B+2
2640 N=C+B+A*L
2650 RETURN
2660 CLS
2670 Q=3
2680 GOSUB 2120
2690 GOSUB 7020
2700 PRINT $ PL. W$;

```

```

2710 GOSUB 6960
2720 GOSUB 2010
2730 IF Q>17 THEN 2660
2740 I=Q
2750 P=O(I, 1)
2760 IF P=0 THEN GOSUB 7020: PRINT # PL, E$:: GOSUB 6960: GOTO 2820
2770 IF(I=16 AND Z=1)OR(I=6 AND Z=2)OR(I=4 AND Z=3)OR(I=3 AND Z=4)THEN
  2900
2780 IF I=7 AND DF=1 THEN 3320
2790 GOSUB 7020
2800 PRINT # PL, O$(I):" - no effect";
2810 GOSUB 6960
2820 GOSUB 7020
2830 PRINT # PL,"something else?";
2840 GOSUB 6930
2850 A$=INKEY$
2860 IF A$="" THEN 2850
2870 IF A$="Y" THEN 2660
2880 GOSUB 6610
2890 RETURN
2900 GOSUB 6610
2910 GOSUB 7020
2920 PRINT # PL,"you're in a ";MID$(O$(Z), 4);
2930 GOSUB 6930
2940 J=0
2950 GOSUB 7020
2960 PRINT # PL,"SEARCH or LEAVE?";
2970 A$=INKEY$
2980 IF A$="" THEN 2970
2990 IF A$="S" THEN J=J+10: GOTO 3020
3000 GOSUB 2590
3010 RETURN
3020 SF=0
3030 IF RAND(100)>90 THEN SF=Z: SU=1: Z=9: GOTO 4110
3040 IF RAND(100)>J THEN PRINT # PL,"there's nothing here!"; GOSUB 6930
3050 GOSUB 7020
3060 PRINT # PL,"there is a CHEST here!";
3070 GOSUB 6930
3080 IF RAND(100)>90 THEN 3260
3090 IF D(AL, 4)=-1 THEN GOSUB 7020: PRINT # PL,"it's empty!"; GOSUB
  6930: GOTO 3200
3100 D(AL, 4)=-1
3110 GOSUB 7020
3120 PRINT # PL,"containing";
3130 IF O(16, 1)OR RAND(100)>20 THEN 3170
3140 PRINT" a ";O$(16);
3150 O(16, 1)=1
3160 GOTO 3200
3170 J=AND(500)+50
3180 PRINT J, O$(17);
3190 O(17, 1)=O(17, 1)+J
3200 GOSUB 6960
3210 GOSUB 7020
3220 PRINT # PL,"Better leave now";
3230 GOSUB 6960
3240 Q=0
3250 GOTO 3000
3260 J=AND(2)
3270 IF M>20 THEN J=AND(5)
3280 GOSUB 7020
3290 PRINT # PL,"It's a TRAP" J"men are killed";

```

```

3300 M=M-J
3310 GOTO 3200
3320 GOSUB 7020
3330 IF AND(100)<80 THEN PRINT $ PL,"The rope broke-it's useless now";
: GOSUB 6960: O(7, 1)=O(7, 1)-1: DF=0 ELSE PRINT $ PL,"You saved them!";
: GOSUB 6960: DF=0: M=M+2
3340 GOSUB 6610
3350 RETURN
3360 IF M>0 THEN RETURN
3370 PRINT $ PL," ALL YOUR MEN ARE DEAD!"
3380 END
3390 IF TW<2400 THEN GOSUB 7020: PRINT $ PL,"Your army is not strong enough";
: GOSUB 6990: GOSUB 7020: PRINT $ PL,"to make succes likely!";
: GOSUB 6990
3400 IF O(1, 1)=0 THEN GOSUB 7020: PRINT $ PL,"The walls are to high";
: GOSUB 6990: PRINT $ PL,"for you to besiege the tower!"; GOSUB 6990
3410 GOSUB 7020
3420 PRINT $ PL, D$;
3430 GOSUB 6930
3440 PRINT $ PL,"Attack or Leave?";
3450 A$="LA"
3460 GOSUB 1940
3470 IF Q=0 THEN 3450
3480 IF Q=2 THEN 3520
3490 IF AND(100)>70 THEN I=AND(M/10)M=M-I: PRINT $ PL,"A volley of arrows";
: GOSUB 6930: GOSUB 7020: PRINT $ PL,"from the tower has killed";
: GOSUB 6990: GOSUB 7020: PRINT $ PL, I"of your men!"; GOSUB 6990
3500 A=A-I
3510 GOTO 1300
3520 FA=1
3530 BM=AND(40)+160
3540 IW=20
3550 BW=2000
3560 SU=0
3570 GOTO 4500
3580 I=TW/M
3590 FOR SP=1 TO 9
3600 IF 10*SP>=I THEN 3620
3610 NEXT
3620 SP=10-SP
3630 RETURN
3640 AL=I
3650 P=D(I, 1)
3660 FOR Z=1 TO 9
3670 IF P<>VAL(D$(Z)) THEN NEXT: GOTO 880
3680 SU=0
3690 IF Z>5 THEN 4110
3700 GOSUB 7020
3710 PRINT $ PL,"you have come to a ";MID$(D$(Z), 4);
3720 GOSUB 6930
3730 ON Z GOTO 3740, 3790, 3850, 3880, 3910
3740 PRINT $ PL,"the way is barred by a stout";
3750 GOSUB 6960
3760 PRINT $ PL,"door with an iron lock!";
3770 GOSUB 6960
3780 GOTO 880
3790 PRINT $ PL,"the entrance is filled with";
3800 GOSUB 6960
3810 GOSUB 7020
3820 PRINT $ PL,"drifted sand!";
3830 GOSUB 6960

```

```

3840 GOTO 880
3850 PRINT $ PL."it's to dark to see!";
3860 GOSUB 6960
3870 GOTO 880
3880 PRINT $ PL."there is no visible entrance!";
3890 GOSUB 6960
3900 GOTO 880
3910 IF MC<6 THEN 4030
3920 MC=0
3930 FOR I=1 TO 15
3940 O(I, 5)=0
3950 IF RAND(100)>50 OR(I=1 AND A>8) THEN IF I<15 THEN 4020
3960 O(I, 5)=RAND(20)
3970 V=O(I, 3)
3980 GOSUB 6590
3990 O(I, 6)=V
4000 IF I=1 THEN O(I, 5)=1
4010 IF I=15 THEN O(I, 5)=RAND(100)+M
4020 NEXT I
4030 GOSUB 7020
4040 PRINT $ PL."Leave.Buy at market.Inventory?";
4050 A$="LBMI"
4060 GOSUB 1940
4070 IF Q=0 THEN 4030
4080 IF Q=1 THEN GOSUB 2590: GOSUB 6610: GOTO 880
4090 IF Q=4 THEN GOSUB 2090: GOTO 4030
4100 GOTO 5360
4110 GOSUB 150
4120 '
4130 GOSUB 7120
4140 '
4150 GOSUB 7020
4160 PRINT $ PL."You have encountered a band of";
4170 GOSUB 6930
4180 GOSUB 7020
4190 PRINT $ PL, MID$(D$(Z), 4);
4200 GOSUB 6930
4210 BM=RAND(M+M/4)+INT(EG/10)
4220 BS=RAND(10)
4230 BX=RAND(INT(EG/10))
4240 SB=RAND(10)
4250 BI=BM
4260 BX=INT(BX)
4270 IF Z<8 THEN GOSUB 7020: PRINT $ PL,"they have an ESCORT!";: GOSUB
6930
4280 IW=RAND(8)-10*(EX>2)
4290 BW=BM*IW
4300 GOSUB 7020
4310 PRINT $ PL,"There are" BM",";
4320 IF IW<6 THEN PRINT" LIGHTLY"; ELSE IF I>5 AND IW<15 THEN PRINT" W
ELL"; ELSE PRINT"HEAVILY";
4330 PRINT" armed";
4340 GOSUB 6930
4350 IF SU THEN 4470
4360 GOSUB 7020
4370 PRINT $ PL, D$;
4380 GOSUB 6930
4390 PRINT $ PL,"Attack.Run or be Friendly?";
4400 GOSUB 6930
4410 A$="ARF"
4420 GOSUB 1940

```

```

4430 ON Q GOTO 4450, 4860, 5020
4440 GOTO 4360
4450 SU=-1
4460 IF AND(100)>.50 THEN SU=0: GOTO 4500
4470 GOSUB 7020
4480 B$="they"
4490 IF SU=1 THEN B$="you": PRINT $ PL. B$ "were surprised": GOSUB 6
930
4500 F1=1
4510 GOSUB 2150
4520 BW=BM*IW
4530 CLS
4540 PRINT TAB(12)"You";TAB(22)"Them"
4550 PRINT"Men" TAB(12)M: TAB(22)BM
4560 PRINT"Weapons";TAB(12)WV: TAB(22)BW
4570 PRINT"Experience";TAB(12)EX: TAB(22)BX
4580 PRINT $ 400,"LUCK";
4590 GOSUB 150
4600 '
4610 I=AND(100)
4620 J=AND(100)
4630 PRINT $ 415. I" " J:
4640 A$=INKEY$
4650 IF A$="" THEN 4590
4660 I=I-SU*10
4670 J=J+SU*10
4680 TW=WV+I+ST*5+EX*5
4690 IF FA THEN IF O(1, 1)=0 THEN TW=TW/5
4700 WT=BW+J+SB*2+BX*2
4710 M=M+INT((WT>TW)*AND(M/10)+(TW>WT)*((AND(M/20)-1)))
4720 BM=BM+INT((TW>WT)*AND(BW/10)+(WT>TW)*((AND(BM/20)-1)))
4730 IF BM<0 THEN BM=0
4740 GOSUB 3360
4750 PRINT
4760 PRINT"Survivors":TAB(12)M: TAB(22)BM
4770 IF BM=0 THEN 5240
4780 IF BM<B1/4 AND AND(100)>50 AND FA=0 THEN 5050
4790 PRINT
4800 PRINT"Fight, Run or Terms?"
4810 A$="FRT"
4820 GOSUB 1940
4830 IF Q=0 THEN 4790
4840 IF Q=2 THEN 4860
4850 IF Q=3 THEN 4920 ELSE 4500
4860 IF SB>B OR AND(100)>90 OR FA=1 THEN GOSUB 7020: PRINT $ PL."You
got away!": GOSUB 6930: FA=0: GOTO 5150
4870 GOSUB 7020
4880 PRINT $ PL."They caught you!":
4890 GOSUB 6930
4900 SU=1
4910 GOTO 4470
4920 I=WT*AND(100)
4930 J=TW*AND(10)*CH
4940 GOSUB 6930
4950 IF J>I AND FA=0 THEN 5000
4960 PRINT $ PL. R$:
4970 GOSUB 6930
4980 SU=1
4990 GOTO 4470
5000 BM=AND(BM)
5010 GOTO 5050
5020 GOSUB 7020

```

```

5030 IF AND(10)>CH THEN PRINT $ PL, R$;: GOSUB 6960: SU=1: GOTO 4470
5040 BM=AND(BM)
5050 PRINT $ PL, BM"will join you!";
5060 GOSUB 6930
5070 M=M+BM
5080 PRINT
5090 PRINT"They have:"
5100 PRINT
5110 FOR I=8 TO 15
5120 IF AND(100)>50 THEN QT=AND(BM): O(1, 1)=O(1, 1)+QT: PRINT QT: O$(
I)
5130 NEXT
5140 IF AND(100)>90 THEN PRINT"Too bad they had no GOLD!" ELSE QT=INT(
AND(5)*((EG/10)+1)*BM): O(17, 1)=O(17, 1)+QT: PRINT QT: O$(17)
5150 EX=EX+1
5160 PRINT
5170 PRINT
5180 PRINT $ PL,"READY TO GO ON?";
5190 A$="A"
5200 GOSUB 1940
5210 IF SF THEN Z=SF: GOTO 2900
5220 GOSUB 6610
5230 GOTO 1780
5240 IF FA THEN 6820
5250 FOR J=1 TO 2
5260 GOSUB 150
5270 *
5280 GOSUB 7120
5290 *
5300 GOSUB 7120
5310 NEXT
5320 PRINT"YOU WON!"
5330 BM=2
5340 PRINT"You managed to salvage:"
5350 GOTO 5110
5360 HC=0
5370 OC=0
5380 ZL=0
5390 CLS
5400 VF=1
5410 PRINT"ITEM PRICE WT. QUANT."
5420 PRINT
5430 FOR I=1 TO 15:COLOUR 3
5440 IF O(I, 5)=0 THEN 5500
5450 PRINT MID$(STAR$(I), 2)" ";
5460 IF I<10 THEN PRINT" ";
5470 PRINT O$(I); TAB(18)O(I, 6); TAB(23)O(I, 2); TAB(27)O(I, 5)
5480 ZL=ZL+1
5490 IF ZL=7 THEN GOSUB 6480:IF ZL=99 THEN 5510
5500 NEXT I
5510 HB=0
5520 F1=1
5530 GOSUB 2150
5540 PRINT"You have" O(17, 1)"Gold Coins"
5550 PRINT"Your men are carrying" TW"lbs"
5560 GOSUB 7020
5570 PRINT $ PL, W$;
5580 GOSUB 6960
5590 GOSUB 2010
5600 IF Q>17 THEN 5560
5610 V=O(Q, 6)

```

```

5620 OL=3+AND(4)-(5*(V>150))
5630 GOSUB 6590
5640 X=Q
5650 GOSUB 7020
5660 IF O(X, 5)=0 THEN PRINT $ PL, E$:: GOSUB 6960: GOTO 4030
5670 GOSUB 7020
5680 PRINT $ PL, "Your offer?";
5690 GOSUB 6960
5700 GOSUB 2010
5710 Q=INT(Q)
5720 GOSUB 7020
5730 IF Q>V THEN V=Q: GOTO 5930
5740 IF RND(30)=15 THEN V=Q: GOTO 5930
5750 IF RND(20)=12 THEN OL=OL+10
5760 IF RND(100)=74 THEN V=V*2
5770 IF Q<V/3 THEN PRINT $ PL, "not interested!": GOTO 6050
5780 IF Q=HB THEN HC=HC+1
5790 IF Q<HB OR(Q=HB AND HC=2) THEN 6030
5800 HB=Q
5810 PRINT $ PL, "Tell you what I'm going to do--";
5820 GOSUB 6930
5830 ZZ=INT(V*((RND(15)+5)*.01))
5840 IF ZZ<1 THEN ZZ=1
5850 V=V-ZZ
5860 IF V<HB THEN V=HB
5870 IF Q>V THEN V=Q: GOTO 5930
5880 GOSUB 7020
5890 IF RND(100)>50 THEN PRINT $ PL, "For you I'll make it" V"OK?": ELSE
5900 PRINT $ PL, "What do you say to" V"?";
5910 AS=INKEY$
5920 IF AS="" THEN 5900
5930 IF AS<>"Y" THEN OC=OC+1: IF OC<OL THEN 5670 ELSE 6030
5940 GOSUB 7020
5950 PRINT $ PL, H$:
5960 GOSUB 6960
5970 GOSUB 2010
5980 IF Q>O(X, 5) THEN PRINT $ PL, M$: GOSUB 6960: GOTO 5930
5990 IF Q<V/10(17, 1) THEN GOSUB 7020: PRINT $ PL, "not enough GOLD!": G
6000 O(X, 1)=O(X, 1)+Q
6010 O(X, 5)=O(X, 5)-Q
6020 GOTO 4030
6030 GOSUB 7020
6040 PRINT $ PL, Y$:
6050 GOSUB 6960
6060 O(X, 6)=INT(O(X, 6)/4)+O(X, 6)
6070 GOTO 4030
6080 PRINT $ PL, D$ "sell anything?";
6090 AS=INKEY$
6100 IF AS="" THEN 6080
6110 IF AS<>"Y" THEN RETURN
6120 GOSUB 7020
6130 PRINT $ PL, W$:
6140 GOSUB 6960
6150 GOSUB 2010
6160 IF Q>17 THEN 6120
6170 I=Q
6180 IF O(I, 1)=0 THEN PRINT $ PL, E$: GOSUB 6960: GOTO 6080
6190 V=INT(O(I, 3)/(AND(3)+2))
6200 OL=5+RND(5)-10+(V>80)
6210 HB=O(I, 3)+1

```



```

6220 HC=0
6230 IF RND(100)>50 THEN PRINT # PL,"I'll give you" V"OK?"; ELSE PRINT
# PL,"Will you take" V"?";
6240 AS=INKEY$
6250 IF AS="" THEN 6240
6260 IF AS="Y" THEN 6400
6270 GOSUB 7020
6280 PRINT # PL,"How much do you want?";
6290 GOSUB 6960
6300 GOSUB 2010
6310 J=Q
6320 IF J=HB THEN HC=HC+1 ELSE HC=0
6330 IF J<=V THEN 6400
6340 IF J>HB OR(J=HB AND HC=2)OR OL=0 THEN PRINT # PL, Y$; GOSUB 6960
: RETURN
6350 HB=J
6360 ZZ=INT(V*( RND(20)+10)*.01)
6370 IF ZZ<1 THEN ZZ=1
6380 V=V+ZZ
6390 IF V<J THEN 6230
6400 GOSUB 7020
6410 PRINT # PL, H$;
6420 GOSUB 6960
6430 GOSUB 2010
6440 IF Q>O(I, 1)THEN PRINT # PL, M$; GOSUB 6960; GOTO 6400
6450 O(17, 1)=O(17, 1)+V*Q
6460 O(I, 1)=O(I, 1)-Q
6470 RETURN
6480 J=0
6490 GOTO 6580 PRINT"press <RETURN> to continue, or"
6500 PRINT"<SPACE> to buy!"
6510 PRINT
6520 GOTO 6550
6530 RETURN
6540 PRINT"press <RETURN> to continue"
6550 AS=INKEY$
6560 IF AS="" THEN 6550
6570 IF ASC(AS)=32 THEN ZL=99
6580 RETURN
6590 V=V-RND((V/4))
6600 RETURN
6610 CLS
6620 FOR I=0 TO 10
6630 FOR J=0 TO 3
6640 IF RND(10)>5 THEN POKE SL+I+J*L, 130 : POKE SL+I+J*L-21504,6
6650 NEXT J, I
6660 FOR I=0 TO 38 STEP 2
6670 POKE 17728+I, 132 : POKE I-3776, 8
6680 POKE 17729+I, 133 : POKE I-3775, 8
6690 NEXT
6700 FOR I=1 TO 21 STEP 2
6710 FOR J=0 TO 6
6720 IF RND(10)>6 THEN POKE 17426+I+J*L, 128; POKE 17427+I+J*L, 129; P
OKE I+J*L-4078,15 : POKE I+J*L-4077,15
6730 NEXT J, I
6740 POKE 17999, 134; POKE-3505,10
6750 POKE 18000, 135; POKE-3504,10
6760 POKE 18001, 136; POKE-3503,10
6770 POKE C+B+A*L, 131
6780 FOR X=1 TO 32
6790 IF D(X, 4)<1 THEN POKE C+D(X, 2)*L+D(X, 3), D(X, 1) : POKE C+D(X,

```

```

2)*L+D(X,3)-21504,2
6800 NEXT
6810 RETURN
6820 CLS
6830 FOR I=1 TO 5
6840 GOSUB 150
6850 SOUND 4.89
6860 GOSUB 7120
6870 GOSUB 150
6880 SOUND 4.176
6890 GOSUB 7120
6900 NEXT
6910 COLOUR5:PRINT $ 400,"      YOU'VE CONQUERED THE":PRINT"
FORCES":PRINT"      OF THE EVIL!":PRINT"      ALL HAIL THE VICTOR!!!"
6920 GOTO 6820
6930 FOR WT=1 TO 300
6940 NEXT
6950 RETURN
6960 FOR WT=0 TO 600
6970 NEXT
6980 RETURN
6990 FOR WT=1 TO 1000
7000 NEXT
7010 RETURN
7020 PRINT $ PL. STRING$(32, 32):
7030 RETURN
7040 PRINT $ 409."RUN TO MORLOCK'S TOWER";
7050 COLOUR 16
7060 PRINT $ 454."The ultimate";
7070 PRINT $ 493."Adventure Game";
7080 COLOUR 9
7090 PRINT $ 850."Press <RETURN> to play";
7100 COLOUR 16
7110 RETURN
7120 FOR WT=1 TO 25
7130 NEXT
7140 RETURN
7150 CLS
7160 COLOUR 5
7170 PRINT"EINLEITUNG:"
7180 COLOUR 16
7190 PRINT"IN DIESEM ADVENTURESPIEL SIND SIE DER FUEHRER EINER RAUBR
ITTERBANDE UND STREIFEN DURCHS LAND. UM SICH GELD, WAFFEN UN
D NEUE SOLDATEN ZU BESCHAFFEN. UM SCHLIESSLICH IN DEN < MORLOCK'S "
7200 PRINT"TOWER > EINZUDRINGEN. DORT MUESSEN SIE DANN GEGEN DIE KRAE
FTE DES BOESEN KAEMPFFEN UND DIESE BESIEGEN. UM SELBST HERR UEBE
R DAS LAND ZU WERDEN. DAS SPIELIST IN ENGLISCHER SPRACHE GESCHRIEBEN."
7210 PRINT"DA DIESE KUERZER UND PRAEZISER ALS DIE DEUTSCHE IST."
7220 PRINT"WENN MAN DEM SPIEL AUFMERKSAM FOLGT. WERDEN WOHL KEINE P
ROBLEME AUFTRETEN."
7230 PRINT
7240 COLOUR 9
7250 PRINT"<RETURN>"
7260 COLOUR 5
7270 A$=INKEY$
7280 IF A$="" THEN 7270
7290 GOTO 7300
7300 CLS
7310 PRINT"LANDKARTE:"
7320 COLOUR 16
7330 PRINT"DIE LANDKARTE IST WAEREND DER MEISTEN ZEIT SICHTBAR. SIE
ZEIGT AN, WO SICH DEERSPIELER. MIT DEM KLEINEN MAENNCHEN SYMBOLISI

```

```

ERT, GERADE BEFINDET UND WO      DIE VERSCHIEDENEN INTERESSANTEN PUNKTE
"
7340 PRINT"LOKALISIERT SIND. MAN KANN DIE SPIEL-    FIGUR MIT DEN TASTE
N FUER DIE VIER      HIMMELSRICHTUNGEN, ALSO ";
7350 COLOUR 3
7360 PRINT"N";
7370 COLOUR 16
7380 PRINT")ORTH,";
7390 COLOUR 3
7400 PRINT"S";
7410 COLOUR 16
7420 PRINT")OUTH. ";
7430 COLOUR 3
7440 PRINT" W";
7450 COLOUR 16
7460 PRINT")EST UND ";
7470 COLOUR 3
7480 PRINT"E";
7490 COLOUR 16
7500 PRINT")AST BEWEGEN"
7510 PRINT"FOLGENDE OBJEKTE WERDEN DARGESTELLT:    DA IST ZUM EINEN EI
NE GEWELLTE LINIE.    DIE SICH QUER DURCH DIE KARTE ERSTRECHT.SIE STELL
T EINEN FLUSS DAR, DER NUR      MITTELS EINES BOOTES DURCHQUERT WERDEN
KANN. DIE GEPUNKTETEN FLAECHEEN "
7520 PRINT"STELLEN EINEN SUMPF DAR (SWAMP). IN DEM IHRE SOLDATEN EINSI
NKEN KOENNEN, DIE    DANN NUR MITTELS EINES SEILS (ROPE)      GERETTET
WERDEN KOENNEN. IM NORDOEST-"
7530 PRINT"LICHEN TEIL DES GEBIETES BEFINDET SICH EIN GEBIRGE, DAS TE
ILWEISE MIT STILI    SIERTEN BERGEN ANGEDEUTET IST."
7540 COLOUR 9
7550 PRINT"      <RETURN>"
7560 COLOUR 16
7570 AS=INKEY$
7580 IF AS="" THEN 7570
7590 CLS
7600 PRINT"GANZ IM SUEOOSTEN IST EINE BURGANLAGE    SICHTBAR. DORT IST"
7610 COLOUR 3
7620 PRINT"<MORLOCK'S TOWER>";
7630 COLOUR 16
7640 PRINT", DAS ENDOZIEL IHRES ABENTEUERS. AUF DER KARTE SIND AUSSERDE
M ZWEI 'k's. DIESE BEZEICHNEN DEN STAND-"
7650 PRINT"ORT VON STAEDTEN, WO SIE VERSCHIEDENE "
7660 PRINT"DINGE EINKAUFEN KOENNEN. BEIM DURCH-    WANDERN DER LANOSCH
AFT ERSCHEINEN IN    IHRER NAEHE MANCHMAL NOCH ANDERE      ZEICHEN,
DIESE HABEN DIE SYMBOLIK:"
7670 PRINT"C = CASLTLE (BURG)                      c = CAVE (HOEHLE OD
ER HAVERNE)      R = RUIN (RUINE)                  T = TOWER
(TURM)                      S = SOLDIERS (SOLDATEN)
O = OGRES (ORGS)          "
7680 PRINT"P = PILGRIMS (PILGER)                    W = WIZZARDS (GELEH
RTE)
7690 COLOUR 9
7700 PRINT
7710 PRINT
7720 PRINT
7730 PRINT"<RETURN>"
7740 COLOUR 5
7750 AS=INKEY$
7760 IF AS="" THEN 7750
7770 CLS
7780 PRINT"STADTBESUCH:"
7790 COLOUR 16

```

```

7800 PRINT"BRINGT MAN SEINE FIGUR MIT EINER STADT (CITY) DECKUNGSGLEICH,
DANN KANN MAN DORT AUF DEN MARKTPLATZ GEHEN. DAS ANGE-BOT IST NICHT
IMMER VOLLSTAENDIG, UND MAN SOLLTE AUF DIE ARTIKEL ACHTEN. "
7810 PRINT"MAN BRAUCHT STEIGEISEN (GRABBLING HOOKS)UM IN EINEN TURM EINDRINGEN
ZU KOENNEN, GENAUSO WIE OELLAMPEN (OILLAMPS) UM IN HOEHLN EINZUDRINGEN.
EIN BOOT WAEERE NOETIG UM DEN FLUSS ZU UEBERQUEREN. UM IN EINE RUINE
EINZUDRINGEN SIND "
7820 PRINT"SCHAUFELN (SHOVELS) NOETIG UND IN EINE BURG KOMMT MAN NUR MIT
EINEM GROSSEN SCHLUESSEL, DEN MAN ABER NUR IM SCHATZ EINES TURMS, EINER
RUINE ODER EINER HOEHLE FINDEN KANN. AUCH IST ES SINNVOLL NAEHRUNGSPAKETE
(RATIONS) ZU KAUFEN. "
7830 PRINT"UM DIE MANNSCHAFT SATT ZU HALTEN. DIE PREISE KANN MAN MIT
IMAGINAEAREN HAENDLER AUSHANDELN, INDEM MAN SICH AN DEM GEFORDERTEN
PREIS ORIENTIERT."
7840 COLOUR 9
7850 PRINT"<RETURN>"
7860 COLOUR 16
7870 AS=INKEY$
7880 IF AS="" THEN 7870
7890 CLS
7900 COLOUR 5
7910 PRINT"INVENTORY:"
7920 COLOUR 16
7930 PRINT"DAS INVENTORY IST EINE UEBERSICHT UEBER ALLE SACHEN, DIE MAN
BESITZT, SOWIE DIE ANZAHL DER EIGENEN SOLDATEN UND DEREN TRAGKRAFT.
WAERHLT MAN DAS INVENTORY IN"
7940 PRINT"OET R WILDNIS. SO KANN MAN UEBER ES ARTIKEL WEGWERFEN, WAS
NOETIG WIRD, WENN DIE SOLDATEN DIE GESAMTE LADUNG NICHT MEHR TRAGEN
KOENNEN."
7950 COLOUR 5
7960 PRINT"EINDRINGEN:"
7970 COLOUR 16
7980 PRINT"WILL MAN IN EINE BURG, EINEN TURM, EINE RUINE ODER IN EINE
HOEHLE EINDRINGEN, SO MUSS MAN SEINE FIGUR DAZU DECKUNGSGLEICHBRINGEN
UND 'U' FUEER 'USE' BENUTZEN DRUECKEN. DANN WIRD DAS INVENTORY
AUFGELISTET ";
7990 PRINT"UND NACH DEN ZU BENUTZENDEN ARTIKEL GEFRAGT. IST DIESER
RICHTIG, SO WIRD DIE KARTE GEZEICHNET UND MAN WIRD GEFRAGT, OB MAN
SUCHEN ODER ODER DEN ORT VERLASSEN WILL. NACH ERFOLG- REICHER
SUCHE KANN MAN ENTWEDER EINEN SCHATZ (CHEST) ";
8000 PRINT"BZW. FEINLICHE SOLDATEN FINDEN"
8010 COLOUR 9
8020 PRINT"<RETURN>"
8030 COLOUR 5
8040 AS=INKEY$
8050 IF AS="" THEN 8040
8060 CLS
8070 PRINT"KAMPF:"
8080 COLOUR 16
8090 PRINT"MAN KANN MIT SOLDATEN ODER DER ESKORTE VON PILGERN ODER
GELEHRTEN KAEMPFEN. BEGEGNET MAN SOLCHEN GIBT DER RECHNER DIE ANZAHL
UND DIE KAMPFSTAERKE (LIGHT, WELL ODER HEAVY) DER GEGNER AN. MAN "
8100 PRINT"HAT DANN DIE WAHL MIT IHNEN ZU KAEMPFEN (ATTACK BZW. FIGHT),
WEGZULAUFEN (RUN) ODER SICH MIT IHNEN ANZUFREUNDEN (FRIENDLY, BZW.
TERMS). DER KAMPF IST BE-ENDET, WENN SIE ODER DER GEGNER VER-
NICHTET SIND, SIE GEFLUECHTET SIND "
8110 PRINT"ODER FEINDE ZU IHNEN UEBERGELAUFEN SIND."
8120 PRINT"<RETURN>"
8130 AS=INKEY$
8140 IF AS="" THEN 8130
8150 GOTO 80

```

8160 POKE -3071, 32
8170 POKE -3070, 112
8180 POKE -3069, 83
8190 POKE -3068, 122
8200 POKE -3067, 255
8210 POKE -3066, 203
8220 POKE -3065, 254
8230 POKE -3064, 64
8240 POKE -3063, 224
8250 POKE -3062, 248
8260 POKE -3061, 184
8270 POKE -3060, 212
8280 POKE -3059, 222
8290 POKE -3058, 187
8300 POKE -3057, 255
8310 POKE -3056, 9
8320 POKE -3055, 72
8330 POKE -3053, 81
8340 POKE -3052, 4
8350 POKE -3051, 73
8360 POKE -3049, 41
8370 POKE -3048, 60
8380 POKE -3047, 188
8390 POKE -3046, 152
8400 POKE -3045, 255
8410 POKE -3044, 61
8420 POKE -3043, 37
8430 POKE -3042, 36
8440 POKE -3041, 36
8450 POKE -3038, 128
8460 POKE -3037, 193
8470 POKE -3036, 227
8480 POKE -3035, 127
8490 POKE -3034, 62
8500 POKE -3033, 28
8510 POKE -3031, 60
8520 POKE -3030, 126
8530 POKE -3029, 255
8540 POKE -3028, 195
8550 POKE -3027, 129
8560 POKE -3023, 146
8570 POKE -3022, 254
8580 POKE -3021, 254
8590 POKE -3020, 84
8600 POKE -3019, 127
8610 POKE -3018, 93
8620 POKE -3017, 95
8630 POKE -3016, 18
8640 POKE -3015, 31
8650 POKE -3014, 31
8660 POKE -3013, 31
8670 POKE -3012, 30
8680 POKE -3011, 255
8690 POKE -3010, 171
8700 POKE -3009, 255
8710 POKE -3008, 73
8720 POKE -3007, 255
8730 POKE -3006, 255
8740 POKE -3005, 255
8750 POKE -3004, 103
8760 POKE -3003, 255
8770 POKE -3002, 249

8780 POKE -3001, 249
8790 RETURN
8800 COLOUR 14
8810 PRINTCHR\$(205);STRING\$(38,202);CHR\$(224);
8820 FOR W=1 TO 21
8830 PRINTCHR\$(202);STRING\$(38,32);CHR\$(202);
8840 NEXT
8850 PRINTCHR\$(228);STRING\$(38,202);CHR\$(226);
8860 COLOUR 15
8870 RETURN

BAR

Gehören Sie auch zu den Leuten, die meinen, man müsse für seinen Computer möglichst viele sinnvolle Anwendungen finden? Nun hier ist eine: Unser Colour-Genie-Programm "Bar" ! Aus fast 80 verschiedenen Drinks-Rezepten sucht das Programm sicher einen Drink heraus, der Ihnen schmeckt. Dabei kann man alle Rezepte ausgeben, bestimmte Drinks nach ihrem Namen suchen, oder als Clou eine Liste der verfügbaren Zutaten eingeben, woraufhin das Colour-Genie alle Rezepte ausgibt, die mit den eingegebenen Zutaten gemixt werden können.

Noch zwei Hinweise zum Eintippen:

- Wir haben das Programm mit 40 Zeichen pro Zeile gelistet, damit Sie mit der Bildschirmformatierung keine Schwierigkeiten haben.
- Lassen Sie sich nicht schon beim Eintippen zum Probieren der Rezepte verleiten, sonst riskieren Sie u.U. etliche Tippfehler.

(Für dieses Programm muß Ihr Colour-Genie 32K RAM haben.)

Programmlisting:

```
10 POKE 16553, 255
20 CLEAR 600
30 DEFINT I, J
40 DEFSTR D, N, M, R, V, X, S, L
50 DIM R(11), M(11), V(50), L(50)
60 ON ERROR GOTO 2860
70 CLS
80 FOR I=1 TO 239
90 COLOUR AND(16)
100 PRINT"BAR ";
110 NEXT
120 PRINT"BAR";
130 FOR I=1 TO 500
140 NEXT I
150 COLOUR 2
160 CLS
170 PRINT @ 405,"DRINKS,DRINKS AND MORE
DRINKS )"
180 FOR I=1 TO 800
190 NEXT I
200 CLS
210 COLOUR 2
220 PRINT @ 5,"***** DRINKS *****
===="
230 PRINT
240 COLOUR 5
250 PRINT"WOLLEN SIE"
260 COLOUR 5
```

```

270 PRINT
280 PRINT"1) AUFLISTUNG ALLER VORHANDENE
N REZEPTE"
290 PRINT"2) WISSEN WAS SIE MIT IHREN ZU
TATEN          MIXEN KOENNEN"
300 PRINT"3) DAS REZEPT FUER EINEN GANZ
BESTIMMTEN DRINK
310 COLOUR 2
320 PRINT
330 PRINT"FUER DIE RUECKKEHR ZU DIESER L
ISTE BITTE ";
340 COLOUR 3
350 PRINT"'R' ";
360 COLOUR 2
370 PRINT"DRUECKEN"
380 D=INKEY$
390 IF D="" THEN 380
400 ID=VAL(D)
410 ON ID GOTO 420, 680, 1740
420 COLOUR 7
430 PRINT
440 PRINT"ZUM ANHALTEN DER AUFLISTUNG BI
TTE          'SHIFT & 9'GLEICHZEITIG DRUECK
EN"
450 FOR I=1 TO 200
460 NEXT
470 RESTORE
480 READ N
490 PRINT
500 COLOUR 16
510 PRINT STRING$(40, 217):
520 PRINT N
530 COLOUR 5
540 FOR I=1 TO 400
550 NEXT
560 READ M
570 PRINT
580 PRINT M
590 PRINT
600 I=0
610 RE=INKEY$
620 IF RE="R" THEN RE="":GOTO 200
630 COLOUR 3
640 READ M(I), R(I)
650 IF LEFT$(R(I), 1)="*" THEN: N=R(I):
GOTO 490
660 PRINT M(I), R(I)
670 GOTO 630
680 PRINT
690 COLOUR 14
700 PRINT"BITTE TIPPEN SIE JEWEILS J (=J
A) FUER JEDE ZUTAT EIN, DIE SIE VORRAE
TIG HABEN.ANDERNFALLS 'N' EINTIPPEN."
710 FOR I=0 TO 50
720 V(I)="N"
730 NEXT I
740 I=0

```



```

750 L(0)="ZUCKER"
760 L(1)="ZITRONE"
770 L(2)="ORANGE"
780 L(3)="ANANAS"
790 L(4)="GAAPEFRUIT"
800 L(42)="KIRSCH"
810 L(5)="STAUBZUCKER"
820 L(6)="ANGOSTURA"
830 L(7)="GRENADINE"
840 L(8)="WERMUT"
850 L(9)="GIN"
860 L(10)="RUM"
870 L(11)="WHISKY"
880 L(12)="SEKT"
890 L(13)="CURACAO"
900 L(14)="GRAND MARNIER"
910 L(15)="COINTREAU"
920 L(16)="APRICOT BRANDY"
930 L(17)="CHERRY BRANDY"
940 L(18)="PRINZEN KIRSCH"
950 L(19)="MARASCHINO"
960 L(20)="BANANENLIKOER"
970 L(21)="KOKOSNUSSCREME"
980 L(22)="PFEFFERMINZLIKOER"
990 L(23)="CREME DE KAKAO"
1000 L(24)="KAFFEE LIKOER"
1010 L(25)="CALVADOS"
1020 L(26)="CORDIAL MEDOC"
1030 L(27)="SOUTHERN COMFORT"
1040 L(28)="WODKA"
1050 L(29)="COGNAC"
1060 L(30)="BRANDY"
1070 L(31)="PORT"
1080 L(32)="SHERRY"
1090 L(33)="ROTWEIN"
1100 L(34)="TONIC WASSER"
1110 L(35)="GINGER ALE"
1120 L(36)="ROSES LIME JUICE"
1130 L(37)="TRINKSCHOKOLADE"
1140 L(38)="MOCCA"
1150 L(39)="EI"
1160 L(40)="ZITRONENSORBET"
1170 L(41)="VANILLE EIS"
1180 L(42)="TEQUILA"
1190 L(43)="CASSIS"
1200 L(44)="BENEDICTINE"
1210 L(45)="CAMPARI"
1220 PRINT
1230 FOR J=0 TO 45
1240 COLOUR(J AND 1)+2
1250 PRINT L(J);
1260 PRINT TAB(20); CHR$(14);
1270 X=INKEY$
1280 IF X="" THEN 1270
1290 IF X="J" THEN V(I)=L(J): I=I+1
1300 PRINT X

```

```

1310 NEXT J
1320 PRINT CHR$(15);
1330 PRINT
1340 PRINT"SUCHE BEGINNT - BITTE WARTEN"
1350 RESTORE
1360 READ N
1370 READ M
1380 I=0
1390 READ M(I), R(I)
1400 IF LEFT$(R(I), 1)="*" THEN: NA=R(I)
: R(I)="": GOTO 1430
1410 I=I+1
1420 GOTO 1390
1430 I=0
1440 J=0
1450 RE=INKEY$
1460 IF RE="R" THEN: RE="": GOTO 200
1470 IF LEFT$(R(I), LEN(V(J)))=V(J) THEN
1550
1480 J=J+1
1490 IF V(J)<>"N" THEN 1470
1500 FOR I=0 TO 11
1510 R(I)=" "
1520 NEXT I
1530 N=NA
1540 GOTO 1370
1550 I=I+1
1560 IF R(I)<>" " THEN 1440
1570 PRINT
1580 COLOUR 16
1590 PRINT STRING$(40, 217)
1600 COLOUR 5
1610 PRINT N
1620 PRINT
1630 COLOUR 2
1640 PRINT M
1650 PRINT
1660 I=0
1670 COLOUR 3
1680 PRINT M(I), R(I)
1690 I=I+1
1700 IF R(I)<>" " THEN 1670
1710 FOR I=1 TO 999
1720 NEXT I
1730 GOTO 1500
1740 PRINT
1750 INPUT"WELCHEN DRINK SUCHEN SIE";S
1760 IF RIGHT$(S, 1)=" " THEN: S=LEFT$(S
, LEN(S)-1): GOTO 1760
1770 S="* " +S
1780 RESTORE
1790 FOR I=1 TO LEN(S)
1800 IF MID$(S, I, 1)=" " THEN IF MID$(S
, I+1, 1)=" " THEN: S=LEFT$(S, I)+RIGHT$(
S, LEN(S)-I-1): GOTO 1790

```

```

1810 NEXT
1820 PRINT S
1830 READ D
1840 RE=INKEY$
1850 IF RE="R" THEN: RE="": 40
1860 IF LEFT$(D, LEN(S))<>S THEN 1830
1870 N=D
1880 PRINT
1890 PRINT N
1900 I=0
1910 READ M
1920 PRINT
1930 COLOUR 16
1940 PRINT STRING$(40,217)
1950 PRINT M
1960 PRINT
1970 READ M(I), R(I)
1980 IF LEFT$(R(I), 1)="*" THEN 2040
1990 COLOURS
2000 PRINT M(I),
2010 COLOUR2
2020 PRINT R(I)
2030 GOTO 1970
2040 RE=INKEY$
2050 IF RE<>"R" THEN: RE="": GOTO 2040 E
LSE 200
2060 DATA* RAUHEREIF-COCKTAIL, SHAKER MIT
EIS - ZUTATEN - GUT SCHUETTELN
UND IN EIN COCKTAIL- GLAS MIT ZUCKER
RAND SEIHEN, 1/3, GIN, 1/3, CURACAO TRIP
LE SEC, 1/3, ZITRONENSAFT, 1 SPRITZER, A
UM
2070 DATA 0.* LONG LUCAS, MIXBECHER MIT
EIS - ZUTATEN - MIT SEKT AUFFUELLEN - 1
ORANGEN - SCHEIBE EINLEGEN - MIT
ZITRONENSCHNEIBE GARNIEREN, 1/2, GIN, 1/
2, CURACAO ORANGE, SAFT 1, ORANGE, 1/2,
ZITRONE
2080 DATA 0.* INFERNO, IN EINER SEKTSCHA
LE ZUTATEN MIT ZITRONENSCHALE ABSPRITZEN
- DANN MIT EINEM SCHUSS FRISCHEM O
RANGENSAFT UND SEKT AUFFUELLEN - EISS
TUECKE BEIGEBEN, 1/3, GIN, 2/3, CURACAO
ORANGE, , ORANGENSAFT, , SEKT
2090 DATA 0.* GIN FIZZ, IM SHAKER MIT KL
EINGESTOSSENEM EIS ZUTATEN KRAEFTIG
SCHUETTELN IN EIN GLASGEBEN UND MIT SO
DA AUFSPRITZEN, 2/3, GIN, 1/3, ZITRONENS
AFT, 1 TL, STAUBZUCKER
2100 DATA 0.* SILVER MOON, ZUTATEN MIT E
IS MIXEN UND EINER HALBEN SCHEIBE ORANG
E GARNIEREN, 1/3, GIN, 1/3, APRICOT BRAN
DY, 1/3, ORANGENSAFT
2110 DATA 0.* MARTINI DRY, ZUTATEN MIT E
ISWUERFELN IM MIXBECHER GUTVERAUEHREN; D
URCHSEIHEN - IM COCKTAILGLAS
MIT OLIVE SERVIEREN - NACH WUNSCH MIT ZI

```

TRONENSCHALE ABSPRITZEN, 2 DASH, ORANGE-
 BITTER, 1/2, WERMUT DRY, 1/2, GIN
 2120 DATA 0.* WHITE LADY, SHAKER MIT EIS
 - ZUTATEN - GUT SCHUETTTELN UND
 IN EIN COCKTAIL- GLAS SEIHEN, 3
 /6, GIN, 2/6, CURACAO TRIPLE SEC, 1/6, Z
 ITRONENSAFT
 2130 DATA 0.* LUCAS' LEMON SPECIAL, ZUTA
 TEN MIT 2 EL GESTOSSENEM EIS INS LONG
 DRINKGLAS - MIT MINERALWASSER AUFF
 UELLEN, 1 GLAS, ZITRONE, 1 GLAS, GIN, 2
 GLAESER, GRAPEFRUIT-SAFT, 1, KIRSCH
 2140 DATA 0.* ADRIA LOOK, ZUTATEN IM MIX
 BECHER KURZ SCHUETTTELN; INEINE SEKTSCHAL
 E GIESSEN UND MIT SEKT AUFFUELLEN, 2 CL,
 GIN, 2 CL, CURACAO, 2 TL, ZITRONENSAFT
 (NATUR), , SEKT ZUM AUFFUELLEN
 2150 DATA 0.* GIN TONIC, ZUTATEN AUF EIS
 GEBEN UND MIT TONIC AUFGIESSEN, 1 G
 LAS, GIN, 1 SCHEIBE, ZITRONE, , TONIC WA
 SSSER ZUM AUFFUELLEN
 2160 DATA 0.* MOON DRIVER, ZUTATEN MIT E
 IS MIXEN- IN SEKTSCHALE GEBEN- MIT SE
 KT AUFFUELLEN GARNIERUNG 1 MARASCHINO-
 KIRSCH + 1 SCHEIBE ZITRONE, 3 CL, GIN,
 2 CL, PRINZEN KIRSCH, 1 CL, WERMUT, , SE
 KT ZUM AUFFUELLEN
 2170 DATA 0.* BOLS FROST, IM TUMBLER EIS
 MIT ZUTATEN VORBEREITEN- MIT TONIC AUFF
 UELLEN GARNIERUNG SCHEIBE ZITRON
 E- ORANGE- KIRSCH, 4 CL, GIN, 2 CL, CUR
 ACAO, 2 CL, ZITRONENSAFT, , TONIC WASSER
 ZUM AUFFUELLEN
 2180 DATA 0.* SILVER TOP, ZUTATEN IM MIX
 BECHER MIXEN- MIT SEKT IN SEKTSCHALE AUF
 FUELLEN GARNIERUNG SCHEIBE ORANGE
 - KIRSCH, 1/3, CURACAO TRIPLE SEC, 1/3,
 GIN, 1/6, GRENADINE, 1/6, ORANGENSAFT,
 , SEKT ZUM AUFFUELLEN
 2190 DATA 0.* BOLS TOP SLING, ZUTATEN IN
 MIXBECHER - 2-3 STUECKE EIS IN TUMBLER
 ; UMRUEHREN UND MIT TONIC AUFFUELLEN
 , 4 CL, GIN, 1/2 CL, GRENADINE, 1/2 CL,
 CURACAO, A DASH, ANGOSTURA, 2, KIRSCHEN,
 1/2 SCHEIBE, ORANGE
 2200 DATA 1/2 SCHEIBE, ZITRONE, , TONIC
 WASSER
 2210 DATA 0.* HONOLULU COOLER, ZUTATEN M
 IXEN IN AUSGEHOEHLTER ANANAS SERVIEREN
 , 1/3, ANANASSAFT, 1/3, ROSES LIME JUICE
 , 1/3, SOUTHERN COMFORT
 2220 DATA 0.* PINA COLADA, ZUTATEN MIXEN
 - VERSCHIEDENE RUMSORTEN -UND IN KOKOSN
 USS SERVIEREN, 1/4 L, ANANASSAFT,

2 EL. KOKOSNUSSCREME. 1 SCHUSS. COINTREA
 U. VIEL. RUM
 2230 DATA 0.* MAI-TAI. ZUTATEN KRAEFTIG
 SCHUETTeln - GARNIERUNG ZITRONENSCHNEIBE +
 KIRSCHEN. 1/2. ZITRONENSAFT. 1/2. RO
 SES LIME JUICE. 1 SCHUSS. CURACAO WEISS,
 2 CL. RUM WEISS. 2 CL. RUM BRAUN
 2240 DATA 0.* PLANTER'S PUNCH. ZUTATEN I
 N EINEM HOHEN GLAS MIXEN UND AVIEREN.
 1/3. ORANGENSAFT. 1/3. ZITRONENSAFT. 1/3
 . ANANASSAFT. 1 DASH. GRENADINE. 2 CL. R
 UM WEISS. 2 CL. RUM BRAUN
 2250 DATA 0.* MINT-FRAPPE. HOHES GLAS ZU
 R HAELFTE MIT GESTOSSENEM EIS FUELLEN -
 LIKOER DARUEBER LAUFEN LASSEN - MIT
 SODAWASSER AUFFUELLEN GARNIERUNG FR
 ISCHE MINZE. BELIEBIG. PFEFFERMINZLIKOER
 2260 DATA 0.* PUSSY-FOOT. EIS + ZUTATEN
 MIXEN - MIT FRUECHTEN GARNIEREN. 1/3
 . ANANASSAFT. 1/3. ORANGENSAFT. 1/3. GRA
 PEFRUITSAFT
 2270 DATA 0.* SUMMERWIND. ZUTATEN + 3 EI
 SWERFEL MIXEN - MIT BITTER GRAPEFR
 UIT AUFFUELLEN. 1 TL. ZITRONENSAFT. 2 G
 LAESER. WODKA. , BITTER GRAPEFRUIT
 2280 DATA 0.* BANANA DAIQUIRI. IM ELEKTR
 ISCHEN MIXER EIS + EINE HALBE BANANE MI
 T ZUTATEN MIXEN - BIS EIS ZU SCHNEE WI
 RD - IM TUMBLER SERVIEREN. 2 CL. BANANEN
 LIKOER. 4 CL. RUM BRAUN. 1 DASH. ZITRON
 E
 2290 DATA 0.* CREOLE. IM SHAKER ZUTATEN
 + EIS + TRINKSCHOKOLADE MI
 XEN MIT SCHOKOLADEN- STREUSEL SERVIEREN
 . 2 CL. CREME DE KAKAO. 4 CL. RUM BRAUN
 . BELIEBIG. TRINKSCHOKOLADE
 2300 DATA 0.* BRASIL. ZUTATEN + EIS IN E
 ISBECHER GEBEN - UMRUEHREN - SERVIE
 REN. 8 CL. MOCCA STARK. 1 EL. ZUCKER
 BRAUN. 4 CL. RUM
 2310 DATA 0.* ZOMBIE. ZUTATEN IN LOGORIN
 KGLAS GEBEN + FRUECHTE + EIS VOR
 SICHTIG UMRUEHREN MIT ZIMTSTANGE SER
 VIEREN. 2 CL. KAFFEELIHOER. 4 CL. RUM B
 RAUN
 2320 DATA 0.* PRAIRIE OYSTER. NACHEINAND
 ER IN EIN COCKTAILGLAS GEBEN. 1 TL. WORC
 ESTERSAUCE. 2 TL. KETCHUP. TROPFEN. OEL.
 TROPFEN. ESSIG. 1. EIGELB. , SALZ. , PF
 EFFER. , CURRY. , TABASCO. , PAPRIKA. ,
 MUSKATNUSS
 2330 DATA 0.* PORTWEIN COBBLER. HOHES GL
 AS ZU 2/3 MIT GESTOSSENEM EIS FUELLEN
 - ZUTATEN - MIT SODAWASSER AUFFUELLEN
 MIT FRUECHTEN GARNIEREN. 2 TEILE. PORT.

1 TEIL, GRAND MARNIER, 1 TEIL, MARASCHI
 NO, 1 TL, ZUCKERSIRUP
 2340 DATA 0.* SHERRY-COBBLER, COBBLER GL
 AS ZU 1/3 MIT FEINEM EIS FUELLEN -
 ZUTATEN - MIT SODA-WASSER AUFFUELLEN -
 DURCHRUEHREN, 2/3, SHERRY FINO, 1/3, CO
 GNAC, 2 SCHUSS, PORT
 2350 DATA 0.* RUM CRUSTA, ZUTATEN + 5 ST
 UECKE EIS IN SHAKER - 1/2 MINUTE KRAEFTI
 G SCHUETTTELN - IN GLAS MIT ZUCKE
 AKRAUSTE SEIHEN, SAFT 1/2, ZITRONE, 2 EL,
 ZUCKERSIRUP, 2 DASH, ANGOSTURA, 3 DASH,
 MARASCHINO, 1 GLAS, RUM
 2360 DATA 0.* BRANDY COOLER, ZUTATEN + E
 IS IN SHAKER - MIXEN - ANSCHLIESSE
 ND IN WHISKY-BECHER SEIHEN UND MIT GIN
 GERALE AUFFUELLEN, 1 TL, ZUCKER, SAFT 1/
 2, ZITRONE, 1/2 GLAS, BRANDY, , GINGER A
 LE
 2370 DATA 0.* GIN COOLER, WIE BRANDY COO
 LER - JEDOCH GIN STATT BRANDY, ,
 2380 DATA 0.* EYEBRIGHT COOLER, GROSSER
 WHISKYBECHER - ZUTATEN + 3 STUECK
 E EIS - MIT GINGER ALE AUFFUELL
 EN - MIT KIRSCHEN UND ANANAS SERVIERE
 N, SAFT 1/2, ZITRONE, 1/2 GLAS, CALVADOS
 , 1/2 GLAS, CORDIAL MEDOC, , GINGER ALE
 2390 DATA 0.* CABLEGRAM COOLER, GROSSER
 WHISKYBECHER - ZUTATEN + 3 STUECK
 E EIS - MIT GINGER ALE AUFFUELL
 EN - UMRUEHREN - MIT ZWEI KIRSCHEN
 SERVIEREN, SAFT 1/2, ZITRONE, 1 EL, GRE
 NADINE, 1/2 GLAS, WHISKY, , GINGER ALE
 2400 DATA 0.* LOVER'S DREAM, ZUTATEN IM
 MIXBECHER MIT GABEL DURCHSCHLAG
 EN - MIT GINGER ALE AUFFUELLEN
 - MIT KIRSCHEN UND STROHHALM SERVIEREN,
 1, EL, 1 EL, ZUCKER
 2410 DATA SAFT 1/2, ZITRONE, , GINGER AL
 E
 2420 DATA 0.* CHAMPAGNER DAISY, ZUTATEN I
 N ZU 1/3 MIT EIS GEFUELLTEN MIXBECHER GE
 BEN- 1/2 MINUTE LANG MIXEN - INSEKTSCHAL
 E MIT SEKT AUFFUELLEN - MIT 2 KIRSCHEN
 N UND 2 STUECKEN ANANAS SERVIEREN
 , SAFT 1/4, ZITRONE, 1 TL, GRENADINE
 2430 DATA 1/2 GLAS, BRANDY, , SEKT
 2440 DATA 0.* BRANDY DAISY, WIE CHAMPAGN
 ER DAISY - JEDOCH MIT 1 STATT 1/2 GL
 AS BRANDY - MIT SODA AUFFUELLEN,

2450 DATA 0.* GIN DAISY, WIE CHAMPAGNER
 DAISY - JEDOCH GIN STATT BRANDY, SAFT 1/
 4, ZITRONE, 1 TL, GRENADINE, 1/2 GLAS, G
 IN

2460 DATA 0.* RUM DAISY, WIE CHAMPAGNER
 DAISY - JEDOCH RUM STATT BRANDY, SAFT 1/
 4, ZITRONE, 1 TL, GRENADINE, 1/2 GLAS, R
 UM
 2470 DATA 0.* WHISKY DAISY, WIE CHAMPAGN
 ER DAISY - JEDOCH WHISKY STATT BRANDY
 , SAFT 1/4, ZITRONE, 1 TL, GRENADINE, 1/
 2 GLAS, WHISKY
 2480 DATA 0.* ASCOT DAISY, MIXBECHER ZU
 1/3 MIT EIS FUELLEN - ZUTATEN - SCH
 UETTELN UND IN SEKTSCHALE GEBEN - MIT 5
 KIRSCHEN GARNIEREN - SODAHINZUFUEGEN.
 2 TL, CHERRY BRANDY, SAFT 1/4, ZITRONE,
 2 TL, WHISKY, 2 EL, GIN
 2490 DATA 0.* MANHATTAN, MIXEN - MIT KIR
 SCHE GARNIEREN, 1 DASH, ANGOSTURA, 2/3,
 WHISKY, 1/3, WERMUT ROT
 2500 DATA 0.* BRONX, MIXEN, 1/4, GIN, 1/
 4, WERMUT ROT, 1/4, WERMUT WEISS, 1/4,
 ORANGENSAFT
 2510 DATA 0.* INDIA, MIXEN, 1 DASH, MARA
 SCHINO, 1/3, GRAPEFRUITSAFT, 2/3, GIN
 2520 DATA 0.* MADAME, MIXEN - MIT ZITRON
 ENSPIRALE UND KIRSCHEN GARNIEREN, 1 DASH,
 ANGOSTURA, 2 SCHUSS, WERMUT, 4 CL, GIN
 2530 DATA 0.* MOONLIGHT, MIXEN, 1 DASH,
 ORANGENSAFT, 1 DASH, CURACAO, 1/3, WERMU
 T, 2/3, GIN
 2540 DATA 0.* PARADIES, MIXEN, 1/3, GIN,
 1/3, APRICOT BRANDY, 1/3, ORANGENSAFT
 2550 DATA 0.* EXPRESS, MIXEN, 1/3, GIN,
 1/3, APRICOT BRANDY, 1/3, WERMUT
 2560 DATA 0.* BAMBOO, MIXEN - MIT 1 STUE
 CK ZITRONENSCHALE SERVIEREN, 2 DASH,
 ANGOSTURA, 1/2, SHERRY, 1/2, WERMUT
 2570 DATA 0.* KIR ROYAL, ZUCKER IN SEKTG
 LAS - ZUTATEN DER REIHE NACH BEIGEBEN,
 1 TL, ZUCKER, 1 DASH, ANGOSTURA, 2 CL, C
 OGNAC, , SEKT ZUM AUFFUELLEN
 2580 DATA 0.* OHIO, MIT KALTEM SEKT AUFF
 UELLEN - KIRSCHEN EINLEGEN, 1 DASH, ANGOS
 TURA, 1/3, WHISKY, 1/3, WERMUT ROT, 1/3
 , CURACAO
 2590 DATA 0.* TOM COLLINS, 4 EISWUERFEL
 IN LONGDRINKGLAS - ZUTATEN - MIXEN - MIT
 SODA AUFFUELLEN, SAFT 1, ZITRONE,
 2 TL, ZUCKER, 2 CL, GIN
 2600 DATA 0.* WHISKY FIX, GLAS ZU 1/3 MI
 T EIS FUELLEN - ZUTATEN - UMRUEHREN - ZI
 TRONEN- SCHEIBE UND STROHHALM BEIG
 EBEN, 2 EL, ZUCKERSIRUP, SAFT 1, ZITRONE
 , 1 CL, CURACAO ORANGE, 2 CL, WHISKY SC
 OTCH
 2610 DATA 0.* PRINCE OF WALES, HOWES GLA
 S - 4 EISWUERFEL - ZUTATEN - UMRUEHREN
 UND MIT SEKT AUFFUELLEN - GARNIERUN
 G 1 SCHEIBE ORANGE, 1/2 GLAS, CURACAO,
 2 DASH, ANGOSTURA

2620 DATA 0.* SALTY DOG. GLAS ZUR HAELFT
E MIT EIS FUELLEN - ZUTATEN + 1 PRI
SE SALZ - UMRUEHREN -MIT STROHHALM S
ERVIEREN, 2 CL. WODKA, 4 CL. GRAPEFRUITS
AFT

2630 DATA 0.* PINK CARTER PICK ME UP. SE
KTSCHALE - 2 EISWUERFEL + ZUTATEN - MIT
ZITRONENSCHALE ABSPRITZEN, 3 DASH
. ANGOSTURA, 1/4 GLAS, GRENADINE, 1/4 GL
AS, ZITRONENSAFT, 1/2 GLAS, GIN., SEKT
ZUM AUFFUELLEN

2640 DATA 0.* WOODKA FIZZ. ZUTATEN MIT EI
S GUT SCHUETTELN - IN GLASSEIHEN - MIT S
ODA AUF- FUELLEN, 1/4, WODKA, 3/4. ANANA
SSAFT

2650 DATA 0.* BLOODY MARY. ZUTATEN MIT E
IS SCHUETTELN, . TOMATENSAFTGUT GEKUEHLT
, 2 CL. WODKA, 1 DASH, WORCESTER-SAUCE,
, PFEFFER, . SALZ, . PAPRIKA, 1 DASH, ZI
TRONENSAFT

2660 DATA 0.* SCREW DRIVER. WODKA IN EIN
GLAS MIT EIS GEBEN - MIT SAFT AUFFUEL
LEN - SCHUETTELN, 2 CL. WODKA, .
ORANGENSAFT ZUM AUFFUELLEN

2670 DATA 0.* MOSCOW MULE. ZUTATEN + EIS
INS GLAS - MIT GINGER AUF-FUELLEN, 4 CL
. WODKA, . GINGER ALE ZUM AUFFUELLEN, 1.
ZITRONENSCHLEIBE

2680 DATA 0.* GIMLET. ZUTATEN IM SHAKER
MIT EIS SCHUETTELN. 1/3. ROSES LIME JUIC
E, 2/3, WODKA

2690 DATA 0.*ANGOSTURA SCORPION. MIT EIS
MIXEN, 2 EL. ZUCKERSIRUP, 2 EL. ROSES L
IME JUICE, 1 DASH, ANGOSTURA, 3 EL. RUM

2700 DATA 0.* RUM PUNCH, ZUSAMMENGEHEN -
BITTER & GINGER ALE ALS LETZTES, 1 TASS
E, ZITRONENSAFT, 2 TASSEN, ERDBEERSIRUP,
3 TASSEN, RUM WEISS (JAMAICA), 2 TASS
EN, ANANASSAFT, 1 TASSE, GRAPEFRUITSAFT,
1 TASSE, GINGER ALE, DASHES, ANGOSTURA

2710 DATA 0.* INCIDENT, SORBET + ZUTATEN
IN SCHMALES GLAS GEBEN, 1 EL, ZITRONENS
ORBET, 4 CL. RUM BRAUN EISKALT, 4 CL.
ROTWEIN

2720 DATA 0.* WHITE WITCH, ZUTATEN SCHAU
MIG MIXEN, 1/4, KAKAO RUM LIKOE, 1/4, R
UM, 1/2, VANILLE EIS

2730 DATA 0.* MORGAN'S REVENGE, MIT EIS
MIXEN - MIT SODA AUFFUELLEN, 1/4, RUM, 1
/2, ANANASSAFT, 1/4. ORANGENLIKOE BITT
ER

2740 DATA 0.* PEACH'S SPECIAL. EIS + ZUT
ATEN IM SHAKER MIXEN - MIT KIRSCHES S
ERVIEREN, 1 DASH, ANGOSTURA, 1/3. CHERRY

BRANDY, 1/3. WERMUT WEISS, 1/3. GIN
 2750 DATA 0.* ACAPULCO, 4 ZERKLEINERTE E
 ISWUERFEL IN GLAS - MIT SODA AUFFUELLEN,
 1 TL. ZUCKERSIRUP, 5 CL. TEQUILA, 2.5 C
 L. CASSIS, 1, ZITRONENSCHNEIBE, , SODA ZU
 M AUFFUELLEN
 2760 DATA 0.* TEQUILA CALIENTE, ZUTATEN
 MIXEN & 3 EISWUERFEL MIXEN - MITSODA AUF
 FUELLEN, 4 CL. TEQUILA, 1 CL. CASSIS, 1
 CL. ROSES LIME JUICE, 2 DASHES, GRENADIN
 E, 5-10 CL. SODA
 2770 DATA 0.* THE BLACKEST RUSSIAN, ZUTA
 TEN IN GLAS GEBEN - VORSICHTIG 3 EISWUER
 FEL ZUGEBEN, 2.5 CL. KAFFEELIKOER, 7.5 C
 L. WODKA
 2780 DATA 0.* WERMUT CASSIS, ZUTATEN & 3
 EISWUERFEL IN GLAS GEBEN - MIT SODA AU
 FFUELLEN, 8 CL. WERMUT TROCKEN, 3.5 CL,
 CASSIS
 2790 DATA 0.* WHISKY COCKTAIL, ZUTATEN &
 3 EISWUERFEL IM SHAKER MIXEN, 5 CL. WHI
 SKY, 2 DASHES, ANGUSTURA, 1 TL. ZUCKERSI
 RUP, 1, KIRSCHKE
 2800 DATA 0.* ZUBROWKATINI, ZUTATEN & 3
 EISWUERFEL IM SHAKER MIXEN, 4 CL. WODKA,
 6 CL. PFEFFERMINZLIKOER WEISS, 4 CL. Z
 ITRONENSAFT, 1, KIRSCHKE GRAUEN
 2810 DATA 0.* ADONIS, ZUTATEN & 3 EISWUE
 RFEL VERRUEHREN, 1.5 CL. WERMUT ROT, 3
 CL. SHERRY, 1 DASH, ANGOSTURA
 2820 DATA 0.* HONEYMOON, 3 EISWUERFEL &
 ZUTATEN IM SHAKER MIXEN, 3 DASHES, CURAC
 AO ORANGE, 1 CL. ORANGENSAFT, 1.5 CL. BE
 NEDICTINE, 2.5 CL. CALVADOS
 2830 DATA 0.* RED SHADOW, 3 EISWUERFEL &
 ZUTATEN IM SHAKER MIXEN, 3 DASHES, ANGO
 STURA, 1 CL. APRICOT BRANDY, 1 CL. CHERR
 Y BRANDY, 3 CL. WHISKY
 2840 DATA 0.* TAKE TWO, 3 EISWUERFEL & 2
 UTATEN MIXEN, 2.5 CL. GIN, 1.5 CL. COINT
 REAU, 1 CL. CAMPARI
 2850 DATA 0.* ROLLS ROYCE, ZUTATEN IN GE
 GEBENER REIHENFOLGE IN EIN GLAS GEBEN -
 UMRUEHREN, 2-3. EISWUERFEL, 2.5 CL. GIN,
 1.5 CL. WERMUT DRY, 1 CL. WERMUT WEIS
 S, 1-2 DASHES, BENEDICTINE, 1, KIRSCHKE
 2860 FOR I=1 TO 300
 2870 NEXT I
 2880 RESUME 200

Ein Maschinensprache-Monitor in BASIC

Dieses Programm stammt aus dem 'Colour-Genie-Buch 1'. Es wird in diesem Buch nochmals gelistet, da der Monitor um Routinen zum Disketten-Sektor lesen und schreiben erweitert wurde. Für genaue Erklärungen des Monitors schlagen Sie bitte im 'Colour-Genie-Buch 1' nach, die Erklärung der Diskettenroutinen finden ein paar Seiten weiter.

Nun noch eine Erklärung der einzelnen Monitor-Kommandos:

1. Hex-Dump

Eingabe: Adresse

Ab der eingegebenen Adresse wird der Speicherinhalt in hexadezimal gelistet. Jeweils nach Ausgabe einer Zeile wird die Tastatur auf folgende Tasten abgefragt:

<CLEAR> - Listing abbrechen.

<Leertaste> - Listing anhalten, bis <RETURN> gedrückt wird. Die einzelnen Zeilen haben folgendes Format:

Die ersten 4 Zeichen geben die Adresse an, in der das erste der 8 Bytes, die in dieser Zeile gelistet werden, steht. Dann folgen die 8 Bytes und zum Schluß, etwas abgesetzt, folgt die sogenannte Checksum. Dieses Byte ist das niederwertigste Byte der Summe aller 8 Bytes und dient als Prüfmöglichkeit für die Richtigkeit einer Eingabe. Alle Maschinenprogramme in diesem Buch werden als Hex-Dump mit Prüfsumme gelistet, so daß Sie eine einfache Möglichkeit haben, die Richtigkeit Ihrer Eingabe zu prüfen.

2. ASCII-Dump

Eingabe: Adresse

Ab der eingegebenen Adresse wird der Speicherinhalt als ASCII-Zeichen gelistet. Jede Zeile beginnt mit der Adresse, in der das erste der 24 folgenden Zeichen steht, dann folgen die ASCII-Zeichen. Jedes Byte unter 32 dez. wird als '.' dargestellt, um die Bildschirmausgabe nicht durch Codes wie 'Backspace' etc. zu zerstören. Auch beim ASCII-Dump kann die Ausgabe durch <CLEAR> abgebrochen und durch <Leertaste> angehalten werden.

3. Edit hexadezimal

Eingabe: Adresse

Ab der eingegebenen Adresse kann der Speicherinhalt verändert werden. Es wird jeweils die Adresse und der bisherige Inhalt dieser Speicherstelle ausgegeben. Sie können dann ein neues Byte eingeben, das dann in die Speicherstelle abgelegt wird. Wenn Sie statt einer Eingabe nur <RETURN> drücken, wird der Editmodus abgebrochen.

4. Edit ASCII

Eingabe: Adresse

Ab der eingegebenen Adresse können Sie Texte direkt in den Speicher eingeben. Die Adresse wird angegeben, ihr Inhalt wird in hexadezimal und ASCII (Code kleiner als 32 = '.') ausgegeben. Sie können nun ein Zeichen eingeben, das an dieser Speicherstelle gespeichert wird. Auch hier wird die Eingabe abgebrochen, wenn Sie nur <RETURN> drücken. Achtung: Um ein Leerzeichen in den Speicher einzugeben, müssen Sie folgendes eingeben: " ". Dies liegt daran, daß der INPUT-Befehl des Colour-Basic alle führenden Leerzeichen ignoriert, eine Eingabe eines Leerzeichens allein also als Leerstring angesehen wird.

5. Go to memory

Eingabe: Adresse byte

Es wird zu einem Programm gesprungen, das bei der eingegebenen Adresse beginnt. Wenn Ihr Programm mit einem CALL 0A7FH beginnt, so können Sie ein Byte in das L-Register übergeben. Wenn Sie Ihr Programm mit einem JP 0A9AH beenden, so wird der Inhalt des L-Registers wieder an das Basic übergeben und der Monitor gibt es hexadezimal aus.

6. Hexadezimal Rechnung

Eingabe: Rzahl1 zahl2

Zahl1, Zahl2, ihre Summe und ihre Differenz werden in hexadezimal und dezimal ausgegeben.

7. SYSTEM-Band schreiben

Eingabe: Wstart ende einsprung

Zusätzlich wird noch der Programmname abgefragt. Dieser Name darf maximal 6 Zeichen lang sein und das erste Zeichen muß ein Buchstabe sein. Dann wird der Bereich von start bis ende im SYSTEM-Band-Format auf Kasette geschrieben. Um dieses Band wieder einzulesen, benutzen Sie den SYSTEM-Befehl des Colour-Basic und geben Sie den von Ihnen gewählten Programmnamen an.

8. Laufwerk selektieren

Eingabe: Laufwerksnummer

Das angegebene Laufwerk (0 (= Laufwerksnummer (= 4) wird angesprochen. Alle Diskettenzugriffe sind nun auf dieses Laufwerk gerichtet.

9. Sektor lesen/schreiben

Eingabe: Adresse disk-relativer-sektor flag

flag=1: Der angegebene Sektor wird ab der angegebenen Adresse in den Speicher gelesen.

flag=2: Die ab der angegebenen Adresse im Speicher stehenden 256 Bytes werden in den angegebenen Sektor geschrieben.

flag=3: Wie flag=2, der beschriebene Sektor wird aber als
Inhaltsverzeichnis-Sektor gekennzeichnet.

Anmerkung: Wenn Sie einen Sektor des Inhaltsverzeichnisses
einlesen, wird 'Disk error No. 6' ausgegeben. Dies ist kein (!)
Fehler, sondern Sie können so diese Sektoren erkennen. Wenn Sie
einen solchen Sektor wieder auf die Diskette schreiben, müssen
Sie flag=3 angeben, sonst ist das Inhaltsverzeichnis nicht mehr
lesbar!!!

Wenn Sie statt Hadresse bzw. Adresse PHadresse bzw. PAdresse
angeben, geht das Listing parallel auf den Drucker.

Programmlisting:

```
10 CLS: CLEAR 1000
20 PRINT "Command";: INPUT A$
30 IF A$="" THEN 20
40 GOSUB 670
50 IF B$="P" THEN PR=1: GOSUB 670 ELSE PR=0
60 C$="HAETGRSDW"
70 FOR A=1 TO LEN(C$): IF B$(A) = MID$(C$, A, 1) THEN NEXT A: GOTO 20
80 ON AGOTO 100, 200, 310, 360, 430, 500, 790, 810, 940
90 GOTO 20
100 GOSUB 690
110 X=Z
120 Z=X: GOSUB 750: PRINT Z$; " ";: IF PR=1 THEN PRINT Z$; " ";
130 C=0: FOR Y=0 TO 7: Z=PEEK(X+Y): C=C+Z: GOSUB 750: PRINT RIGHT$(Z$, 2); " ";: IF
PR=1 THEN PRINT RIGHT$(Z$, 2); " ";
140 NEXT Y: PRINT " ";: IF PR=1 THEN PRINT " ";
150 Z=C: GOSUB 750: PRINT RIGHT$(Z$, 2): IF PR=1 THEN PRINT RIGHT$(Z$, 2)
160 X=X+8
170 HB=PEEK(&HF840): IF HB=2 THEN 20
180 IF HB<>128 THEN 120
190 HB=PEEK(&HF840): IF HB<>1 THEN 190 ELSE 120
200 GOSUB 690
210 X=Z
220 Z=X: GOSUB 750: PRINT Z$; " ";: IF PR=1 THEN PRINT Z$; " ";
230 FOR Y=0 TO 23: Z=PEEK(X+Y)
240 IF Z<32 THEN Z=ASC(" ")
250 PRINT CHR$(Z);: IF PR=1 THEN PRINT CHR$(Z);
260 NEXT Y: PRINT: IF PR=1 THEN PRINT
270 X=X+24
280 HB=PEEK(&HF840): IF HB=2 THEN 20
290 IF HB<>128 THEN 220
300 HB=PEEK(&HF840): IF HB<>1 THEN 300 ELSE 220
310 GOSUB 690
320 X=Z
330 Z=X: GOSUB 750: PRINT Z$; " - ";: Z=PEEK(X): GOSUB 750: PRINT RIGHT$(Z$, 2); "
- ";
340 A$="": INPUT A$: IF A$="" THEN 20
350 GOSUB 690: POKE X, Z AND 255: X=X+1: GOTO 330
360 GOSUB 690
370 X=Z
380 Z=X: GOSUB 750: PRINT Z$; " - ";: Z=PEEK(X): GOSUB 750: PRINT RIGHT$(Z$, 2); "
";
390 IF Z<32 THEN Z=ASC(" ")
400 PRINT CHR$(Z); " - ";
```

```

410 A$="":INPUTA$:IFA$=""THEN20
420 POKEX,ASC(A$):X=X+1:GOTO380
430 GOSUB690
440 Z1=INT(Z/256)AND255:Z2=ZAND255
450 POKE16526,Z2:POKE16527,Z1
460 GOSUB670:IFB$="" THEN460ELSEA$=B$+A$
470 GOSUB690:Z=ZAND255
480 Z=USR(Z)
490 GOSUB750:PRINTRIGHT$(Z$,2):GOTO20
500 GOSUB690:IFZ>0THENX1=ZELSEX1=Z+65536
510 GOSUB670:IFB$="" THEN510ELSEA$=B$+A$
520 GOSUB690:IFZ>0THENX2=ZELSEX2=Z+65536
530 PRINT"  A      B      A+B      A-B"
540 PRINT"  ";Z=X1:GOSUB750:PRINTZ$;"  ";Z=X2:GOSUB750:PRINTZ$;"  ";
550 Z=X1+X2:IFZ>65535THENZ=Z-65536
560 GOSUB750:PRINTZ$;"  ";
570 Z=X1-X2:IFZ<0THENZ=Z+65536
580 GOSUB750:PRINTZ$
590 Z=X1:GOSUB640:PRINTZ$;"  ";Z=X2:GOSUB640:PRINTZ$;"  ";
600 Z=X1+X2:IFZ>65535THENZ=Z-65536
610 GOSUB640:PRINTZ$;"  ";
620 Z=X1-X2:IFZ<0THENZ=Z+65536
630 GOSUB640:PRINTZ$:GOTO20
640 Z$=STR$(Z):Z$=RIGHT$(Z$,LEN(Z$)-1)
650 IFLEN(Z$)<5THENZ$=STRING$(5-LEN(Z$),32)+Z$
660 RETURN
670 B$="":IFLEN(A$)=0THENRETURN
680 B$=LEFT$(A$,1):A$=RIGHT$(A$,LEN(A$)-1):RETURN
690 Z=0
700 GOSUB670:IFB$<"0"ORB$>"F"THEN740
710 IFB$>"9"ANDB$<"A"THEN740
720 IFB$<-"9"THENZ1=ASC(B$)-48ELSEZ1=ASC(B$)-55
730 Z=Z*16+Z1:GOTO700
740 IFZ>32767THENZ=Z-65536
750 Z1=INT(Z/256)AND255:Z2=ZAND255:Z$=""
760 Z3=INT(Z1/16):GOSUB770:Z3=Z1AND15:GOSUB770:Z3=INT(Z2/16):GOSUB770:
Z3=Z2AND15
770 IFZ3<10THENZ$=Z$+CHR$(Z3+48)ELSEZ$=Z$+CHR$(Z3+55)
780 RETURN
790 GOSUB670:IFASC(B$)<48ORASC(B$)>51THEN20
800 POKE&H5A08,ASC(B$)-48:CALLDOC7:GOTO20
810 GOSUB690:AD=Z:GOSUB690:SN=Z:GOSUB690:FL=Z
820 RESTORE
830 READA$:IFA$<>"SECIO"THEN830
840 A$="":FORX=1TO15:READA:A$=A$+CHR$(A):NEXTX:FORX=1TOFL:READR:NEXTX
850 X=VARPTR(A$):X=PEEK(X+1)+256*PEEK(X+2):IFX>32767THENX=X-65536
860 POKE X+1,(AD AND 255):POKE X+2,(INT(AD/256)AND255)
870 POKE X+4,(SN AND 255):POKE X+5,(INT(SN/256)AND255)
880 POKE X+7,(DR AND 255):POKE X+8,(INT(DR/256)AND255)
890 DEFUSR1=X:X=USR1(0):IFX=0THEN20
900 PRINT"Disk Error No.":X:GOTO20
910 DATA SECIO
920 DATA 33,0,0,17,0,0,205,0,0,111,38,0,195,154,10
930 DATA-12433,-12417,-12421
940 RESTORE
950 WR$=""
960 READXX$:IF XX$<>"SYSTEM" THEN 960
970 FORX=1TO127
980 READA:WR$=WR$+CHR$(A)
990 NEXTX:X=VARPTR(WR$):X=PEEK(X+1)+256*PEEK(X+2):IFX>32767THENX=X-655
36
1000 R=X+95:R1=INT(R/256)AND255:R2=RAND255

```

```

1010 POKEX+66,R2:POKEX+67,R1:POKEX+75,R2:POKEX+76,R1
1020 GOSUB1220:GOSUB1260:GOSUB1220:GOSUB1260:GOSUB1220
1030 PRINT"Filename: ";INPUTPN$
1040 FORX1=XTOX+5:POKEX1.32:NEXTX1
1050 FORX1=1TOLN(PN$):POKEX-1+X1,ASC(MID$(PN$,X1,1)):NEXTX1
1060 X=X+6:X1=INT(X/256)AND255:X2=XAND255
1070 POKE16526,X2:POKE16527,X1
1080 ONERRORGOTO1100
1090 DEFUSAO=X
1100 ONERRORGOTO0
1110 ONERRORGOTO1130
1120 CMD""
1130 ONERRORGOTO0
1140 X1=USR(0)
1150 GOTO20
1160 DATA SYSTEM
1170 DATA0.0,0.0,0.0,32.32,32.32,32.32,32.32
1180 DATA42.142,64.17,250.255,25.205,63.2,62.85,205.31,2.6,6.126,35.20
5,31.2,16.249,17.244,255.25,229,221,225,221,94,0,221,86.1,221,110.2,22
1.102.3
1190 DATA175.237,82.35,124.183,40,8.6,0,205,0,0,37,24,244,125.69,183.1
96,0,0,62.120,205,31,2
1200 DATA221,126,4,205,31,2,221,126,5,205,31,2,201
1210 DATA62.60,205,31,2,120,205,31,2,123,205,31,2,79,122,205,31,2,129,
79,26,19,205,31,2,129,79,16,247,195,31,2
1220 GOSUB690
1230 Z1=INT(Z/256)AND255
1240 Z2=ZAND255
1250 POKEX,Z2:POKEX+1,Z1:X=X+2:RETURN
1260 GOSUB670
1270 IFB$="" THEN1260
1280 A$=B$+A$:RETURN

```

Ein Grafikprogramm in Maschinensprache: Lissajou-Figuren

Dieses Programm demonstriert die Möglichkeiten mit Maschinensprache schnelle Grafiken anzulegen.

Es berechnet und zeichnet Lissajou-Figuren in den Grafikspeicher.

Da die Figuren aus zwei phasenverschobenen Sinuskurven entstehen, wurde im Programm (ab &H7200) eine Tabelle mit Sinuswerten, die auf die Grafikauflösung abgestimmt sind angelegt. Damit werden lange Berechnungen vermieden. Aus der Tabelle werden einfach für jeden Punkt zwei Koordinaten entnommen, die einen einstellbaren Abstand voneinander haben.

Geben Sie das Programm mit dem weiter vorne gelisteten Basic-Monitor ein. Wenn Ihr Colour-Genie nur mit 16 K-Byte ausgerüstet ist, müssen Sie vorher bei Mem Size ? mit 28672 antworten.

Nachdem Sie es eingegeben haben, sollten Sie das Programm zunächst auf Kassette oder Diskette abspeichern.

Die Anfangsadresse ist &H7000, Endadresse &H7437 und Startadresse ist &H7090.

Falls Sie einen Monitor mit Disassembler zur Verfügung haben, können Sie die Einzeilheiten des Programmes leicht analysieren, da es modular aufgebaut ist.

Noch ein Wort zur Bedienung:

Mit den Pfeiltasten können Sie das Frequenzverhältnis (Pfeil-oben & Pfeil-unten) bzw. die Geschwindigkeit der Phasenverschiebung einstellen (Pfeil-links & Pfeil-rechts).

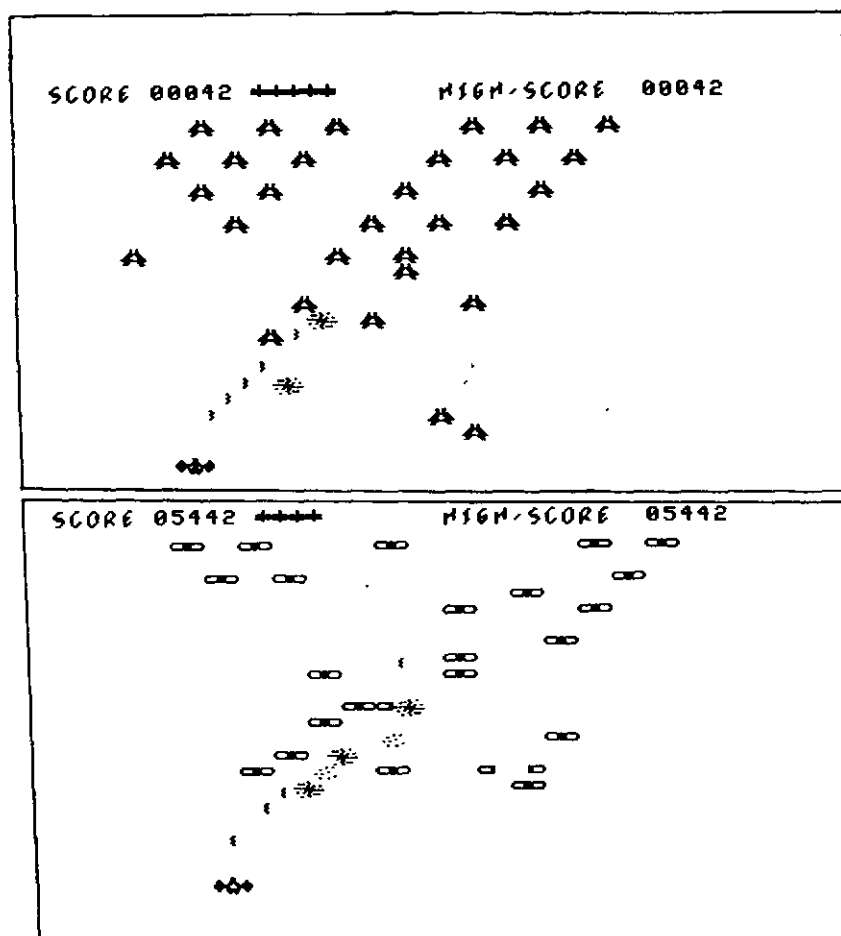
Auf den nächsten drei Seiten finden Sie das Hexlisting des Programms mit Prüfsummen.

7000	E5	D5	21	00	72	5F	16	00	C2
7008	19	7E	D1	E1	C9	00	00	00	12
7010	4F	3A	FF	7F	81	E6	7F	C3	B0
7018	00	70	00	00	00	00	00	00	70
7020	06	7F	C5	78	CD	00	70	C6	C5
7028	20	67	3A	FE	7F	4F	3A	FD	C4
7030	7F	81	32	FE	7F	CD	10	70	FC
7038	6F	CD	C6	3C	C1	05	C2	22	E8
7040	70	C9	00	00	00	00	00	00	39
7048	00	00	00	00	00	00	00	00	00
7050	3A	FF	7F	4F	3A	FC	7F	81	3D
7058	32	FF	7F	CD	20	70	21	FD	2B
7060	7F	3A	40	F8	FE	08	20	01	18
7068	34	FE	10	20	01	35	28	FE	C1
7070	40	20	01	34	FE	20	20	01	D4
7078	35	CD	40	71	C3	D0	70	00	B6
7080	00	00	00	00	00	00	00	00	00
7088	00	00	00	00	00	00	00	00	00
7090	3E	03	32	13	43	3E	02	32	3B
7098	FC	7F	32	FD	7F	21	00	48	92
70A0	11	01	48	01	FF	0F	36	00	9F
70A8	ED	B0	C3	EB	70	00	00	00	B8
70B0	E5	D5	06	05	7E	12	23	C5	3D
70B8	01	28	00	EB	09	EB	C1	10	D9
70C0	F3	D1	E1	C9	00	00	00	00	6E
70C8	00	00	00	00	00	00	00	00	00
70D0	21	00	48	11	00	58	01	48	1B
70D8	0D	ED	B0	21	00	48	11	01	25
70E0	48	01	48	0D	36	00	ED	B0	71
70E8	C3	50	70	3E	18	32	03	43	51
70F0	3E	60	32	09	43	CD	A9	38	CA
70F8	C3	6B	71	00	00	00	00	00	9F
7100	00	A8	88	88	88	A8	20	20	28
7108	20	20	20	A8	08	A8	80	A8	E0
7110	A8	08	28	08	A8	88	88	A8	40
7118	08	08	A8	80	A8	08	A8	A8	38
7120	80	A8	88	A8	A8	08	08	08	18
7128	08	A8	88	A8	88	A8	A8	88	40
7130	A8	08	A8	00	00	00	00	00	58
7138	00	00	00	00	00	00	00	00	00
7140	3A	FC	7F	11	28	48	0E	64	A8
7148	CD	80	71	0E	0A	CD	80	71	94
7150	0E	01	CD	80	71	3A	FD	7F	83
7158	11	40	49	0E	64	CD	80	71	CA
7160	0E	0A	CD	80	71	0E	01	CD	B2
7168	80	71	C9	21	80	72	11	48	26
7170	65	01	FF	01	ED	B0	C3	50	16
7178	70	00	00	00	00	00	00	00	70
7180	D5	21	01	71	11	05	00	19	97
7188	91	30	FC	81	B7	ED	52	D1	05
7190	13	F5	CD	80	70	F1	C9	00	AF
7198	00	00	00	00	00	00	00	00	00
71A0	00	00	00	00	00	00	00	00	00

71A8	00	00	00	00	00	00	00	00	00	00
71B0	00	00	00	00	00	00	00	00	00	00
71B8	00	00	00	00	00	00	00	00	00	00
71C0	00	00	00	00	00	00	00	00	00	00
71C8	00	00	00	00	00	00	00	00	00	00
71D0	00	00	00	00	00	00	00	00	00	00
71D8	00	00	00	00	00	00	00	00	00	00
71E0	00	00	00	00	00	00	00	00	00	00
71E8	00	00	00	00	00	00	00	00	00	00
71F0	00	00	00	00	00	00	00	00	00	00
71F8	00	00	00	00	00	00	00	00	00	00
7200	30	31	33	34	36	37	38	3A	A7	
7208	3B	3D	3E	3F	41	42	43	44	FF	
7210	45	46	47	48	49	4A	4B	4C	44	
7218	4C	4D	4D	4E	4E	4E	4E	4E	6C	
7220	4F	4E	4E	4E	4E	4E	4D	4D	6F	
7228	4C	4C	4B	4A	49	48	47	46	4B	
7230	45	44	43	42	41	3F	3E	3D	09	
7238	3B	3A	38	37	36	34	33	31	B2	
7240	30	2E	2C	2B	29	28	27	25	52	
7248	24	22	21	20	1E	1D	1C	1B	F9	
7250	1A	19	18	17	16	15	14	13	B4	
7258	13	12	12	11	11	11	11	11	8C	
7260	11	11	11	11	11	11	12	12	8A	
7268	13	13	14	15	16	17	18	19	AD	
7270	1A	18	1C	1D	1E	20	21	22	EF	
7278	24	25	27	28	29	2B	2C	2E	46	
7280	00	00	00	00	00	14	00	01	15	
7288	55	40	05	55	00	05	55	00	49	
7290	01	55	00	15	55	50	05	55	6A	
7298	00	50	05	00	01	54	00	40	EA	
72A0	15	00	15	00	15	00	00	00	3F	
72A8	00	00	00	00	00	14	00	01	15	
72B0	55	40	15	55	40	15	55	40	E9	
72B8	05	55	40	15	55	50	15	55	8E	
72C0	40	50	05	00	04	01	01	40	0B	
72C8	51	40	51	40	51	40	00	00	B3	
72D0	01	00	00	01	00	14	00	00	16	
72D8	14	00	54	01	40	54	01	40	3E	
72E0	15	01	50	14	01	50	15	05	E5	
72E8	40	50	05	00	04	51	05	40	2F	
72F0	51	40	51	40	01	40	00	00	63	
72F8	04	40	00	04	40	14	00	00	9C	
7300	14	00	54	00	00	54	00	00	BC	
7308	14	00	50	00	00	50	14	01	C9	
7310	40	50	05	00	04	41	01	40	1B	
7318	15	40	15	00	05	00	00	00	6F	
7320	04	10	00	10	40	14	00	00	78	
7328	14	00	15	55	00	15	55	00	E8	
7330	15	55	50	00	00	50	14	01	1F	
7338	40	50	05	00	04	51	01	40	2B	
7340	01	40	51	40	01	40	00	00	13	

7348	04	04	00	40	40	14	00	00	9C
7350	14	00	05	55	40	05	55	40	48
7358	15	55	50	00	00	50	14	01	1F
7360	40	50	05	00	04	01	01	40	DB
7368	51	40	51	40	51	40	00	00	B3
7370	01	01	01	01	00	14	00	00	18
7378	14	00	00	05	50	00	01	50	BA
7380	14	00	50	00	00	50	14	01	C9
7388	40	50	05	00	01	54	05	50	3F
7390	15	00	15	00	15	00	00	00	3F
7398	01	00	44	01	00	14	00	00	5A
73A0	14	00	00	01	50	00	00	50	B5
73A8	14	00	50	00	00	50	14	01	C9
73B0	40	50	05	00	04	00	00	00	99
73B8	00	00	00	00	00	00	00	00	00
73C0	00	40	10	04	00	14	00	00	68
73C8	14	00	14	05	50	14	01	50	E2
73D0	14	00	50	14	01	50	15	05	E3
73D8	40	54	15	00	05	55	55	55	AD
73E0	55	55	55	55	55	40	00	00	E9
73E8	00	40	44	04	00	15	55	41	33
73F0	55	40	15	55	40	15	55	40	E9
73F8	14	00	50	15	55	50	15	55	88
7400	40	55	55	00	00	00	00	00	EA
7408	00	00	00	00	00	00	00	00	00
7410	00	15	01	50	00	15	55	41	11
7418	55	40	05	55	00	05	55	00	49
7420	14	00	50	05	55	40	05	55	58
7428	00	15	54	00	05	55	55	55	6D
7430	55	55	55	55	55	40	00	00	E9

ZALAGA



**EIN SUPER-MASCHINENSPRACHE-SPIEL
KOMPLETT GELISTET UND ERKLÄRT !**

'ZALAGA' ein Spielprogramm in Maschinensprache fuer das Colour-Genie mit 16KByte RAM.

ZALAGA ist ein Action-Spiel in Maschinensprache, das auf Geschwindigkeit ausgelegt ist.

Es soll einmal demonstrieren, wo die wahren Vorteile der Maschinenprogrammierung liegen.

Sie geben das Programm am guenstigsten mit dem weiter vorne gelisteten Basic-Monitor ein oder mit einem anderen Maschinensprachemonitor. Wenn Ihr Geraet nur mit 16KByte RAM ausgeruestet ist, so geben Sie vorher bei MEM SIZE? 28672 ein und laden dann den Monitor. Das Programm besteht aus zwei grossen Bloecken:

Das eigentliche Programm belegt den Speicher von 7000H bis 75FFH. Es ist wiederum in kleinere Unterprogramme eingeteilt, wie aus der Dokumentation hervorgeht.

Der zweite Teil sind Tabellen. Anfangsgrafik und die Werte fuer die definierbaren Zeichen.

Dieser Block beginnt bei 7800H. Sie haben also noch genugend freien Speicher um das Programm um- oder auszubauen.

Die Luecken zwischen den einzelnen Unterprogrammen erlauben kleine Aenderungen auch innerhalb des ersten Programtteils. Auf jedenfall sollten Sie zuerst eine Original-Version auf Kasette oder Diskette sichern. Also speichern Sie von 7000H bis 7FFFH mit der Startadresse 75B0H.

Noch einige Anmerkungen ueber den Aufbau der Tabellen.

Die Tabelle der Positionen der Feinde besteht aus 128 Bytes. Fuer jedes der 32 Objekte stehen 4 Bytes zur Verfuegung. Die ersten beiden Bytes entsprechen der Position des Feindes im Bildschirmspeicher (4400H-47E7H). Ist diese Position gleich 0, so bedeutet dies, dass der jeweilige Feind schon abgeschossen wurde.

Das 3. Byte beinhaltet das LSB der Y-Richtung des Feindes. Es kann die Werte 00H, 28H oder 08H annehmen. 00H heisst keine Y-Richtung 28H entspricht 40D also 40 Zeichen pro Bewegung mehr (eine Bildschirmzeile) und 08H entspricht -40D also eine Zeile nach oben.

Das dritte Byte dient als Zaehler fuer die Bewegungen eines Feindes nach unten. Nach 24 Bewegungen (da die erste Zeile nicht benutzt wird) ist ein Feind wieder in seiner Ausgangs-zeile.

Die X-Richtung ist fuer alle Feinde immer gleich.

Sobald ein Feind am linken oder rechten Bildschirmrand ist, wird die X-Richtung fuer alle umgekehrt.

Die Tabelle wird, da sie veraendert werden muss, zu Beginn jeder Runde von 7900H nach 7880H kopiert.

Zwei weitere Tabellen sind nach dem gleichen Prinzip aufgebaut.

Die Tabellen der Schuesse und die der Explosionen.

Die Schusstabelle besteht aus 16 Bytes, jeweils LSB und MSB der Schussposition im Bildschirmspeicher. Sie wird ab 7870H angelegt. Die Explosionstabelle besteht aus 24 Bytes ab 7850H. Sie hat zusaetzlich zu der Position der Explosion noch ein Byte als Zaehler fuer die drei Phasen der Explosion. Die Anfangsgrafik beginnt bei 7A00H und belegt 1024 Byte. Die definierbaren Zeichen beginnen bei 7E00 und belegen 512 Bytes. Sie sind zusaetzlich noch einmal vergroessert im Anschluss an das Listing ausgedruckt.

Die ersten 32 Zeichen von 128 bis 159 sind fuer 8 Feindtypen

in je zwei Bewegungszustaenden definiert.
 Danach folgen die Zeichen fuer Schiff, Schuesse, Explosionen usw.
 Doch nun wollen wir Sie nicht weiter vom tippen abhalten.
 Wir wuenschen Ihnen viel Glueck beim Kampf gegen die Zalonen.
 Das Programmlisting:

```

7000 21 00 7E 11 00 F4 01 00 A5
7008 02 ED 80 3E 01 CD 80 3F 9A
7010 CD F0 70 21 00 7A 11 00 D9
7018 44 01 00 04 ED 80 21 00 07
7020 F0 11 01 F0 01 FF 03 36 28
7028 06 ED 80 CD F0 14 2A AA 48
7030 40 7C E6 01 C6 44 67 FE 12
7038 44 28 05 7D FE 18 30 EB 1F
7040 7E FE 7F 28 0A 36 7F 11 F3
7048 00 AC 19 36 02 18 10 E5 0A
7050 D1 EB 01 00 36 09 7E 12 8C
7058 EB 11 00 AC 19 36 05 CD C9
7060 2B 00 FE 30 DA 2B 70 FE CC
7068 3A 38 08 FE 53 C2 28 70 28
7070 C3 80 70 32 3C 47 D6 30 6E
7078 32 00 42 C3 2B 70 00 00 D2
7080 21 00 00 22 02 42 3E 07 CC
7088 32 01 42 3A 3C 47 D6 30 38
7090 32 00 42 3E 07 D3 F8 3E C2
7098 F7 D3 F9 21 E4 79 06 05 4C
70A0 36 80 23 10 FB 3E 05 D3 2A
70A8 F8 3E 03 D3 F9 21 D0 07 FD
70B0 22 0C 42 C3 80 74 00 00 27
70B8 00 00 00 00 00 00 00 00 00
70C0 00 00 00 00 00 00 00 00 00
70C8 00 00 00 00 00 00 00 00 00
70D0 00 00 00 00 00 00 00 00 00
70D8 00 00 00 00 00 00 00 00 00
70E0 00 00 00 00 00 00 00 00 00
70E8 00 00 00 00 00 00 00 00 00
70F0 3E 07 D3 F8 3E FF D3 F9 19
70F8 CD 80 38 C9 00 00 00 00 7E
7100 DD 21 80 78 ED 48 08 42 78
7108 DD 6E 00 DD 66 01 7C B5 C0
7110 CA B3 71 36 20 23 36 20 BD
7118 2B 09 DD 5E 02 16 00 7B 02
7120 FE D8 20 02 16 FF 19 B7 DD
7128 28 0C FE 28 28 05 DD 35 99
7130 03 18 03 DD 34 03 DD 7E 8D
7138 03 B7 20 04 DD 36 02 00 F3
7140 FE 18 20 08 DD 36 02 00 53
7148 DD 36 03 00 11 E8 47 DF 35
7150 38 04 11 40 FC 19 11 28 D8
7158 44 DF 30 04 11 C0 03 19 44
7160 DD 75 00 DD 74 01 3A 01 DF
7168 42 E6 07 07 07 C6 80 5F E2
7170 3A 0A 42 E6 02 83 77 23 8B
7178 3C 77 11 FF AB 19 3A 01 C2
7180 42 3C 77 23 77 DD 7E 03 ED
7188 B7 20 13 CD F0 14 3A 00 F5
7190 42 5F 3A AA 40 BB 30 04 B4
7198 DD 36 02 28 18 15 DD 7E C5
71A0 02 FE 28 20 0E CD F0 14 27
71A8 3A AA 40 FE 01 30 04 DD 34
71B0 36 02 D8 11 04 00 DD 19 1B
71B8 DD 7D B7 C2 04 71 C9 00 11

```

71C0	3E	08	D3	F8	DB	F9	B7	28	C4
71C8	01	3D	D3	F9	21	28	44	11	AB
71D0	28	00	06	16	7E	FE	20	20	00
71D8	05	19	10	F8	18	06	21	01	66
71E0	00	22	08	42	21	4F	44	06	26
71E8	16	7E	FE	20	20	04	19	10	FF
71F0	F8	C9	21	FF	FF	22	08	42	4C
71F8	C9	00	00	00	00	00	00	00	C9
7200	3A	0A	42	3C	32	0A	42	CD	0D
7208	00	71	CD	C0	71	CD	38	74	E8
7210	3A	40	F8	E6	20	28	01	2B	CC
7218	3A	40	F8	E6	40	28	01	23	E4
7220	7D	FE	98	20	01	23	FE	BC	11
7228	20	01	2B	22	06	42	CD	50	D3
7230	74	3A	40	F8	E6	80	28	03	77
7238	CD	70	72	CD	80	73	3A	40	E9
7240	F8	FE	06	CA	00	70	3A	0B	7B
7248	42	B7	C2	00	72	C3	80	74	E4
7250	00	00	00	00	00	00	00	00	00
7258	00	00	00	00	00	00	00	00	00
7260	00	00	00	00	00	00	00	00	00
7268	00	00	00	00	00	00	00	00	00
7270	21	70	78	7E	23	B6	28	07	8F
7278	23	7D	FE	80	20	F5	C9	28	27
7280	EB	2A	06	42	01	D9	FF	09	3F
7288	EB	73	23	72	EB	36	A6	11	CB
7290	00	AC	19	36	03	C9	00	00	C7
7298	E5	FD	21	50	78	FD	6E	00	36
72A0	FD	66	01	7C	B5	28	12	FD	CC
72A8	23	FD	23	FD	23	FD	7D	FE	DB
72B0	68	C2	9D	72	E1	21	00	00	38
72B8	C9	E1	7E	E6	01	28	01	2B	63
72C0	FD	75	00	FD	74	01	FD	36	17
72C8	02	00	EB	FD	21	80	78	FD	00
72D0	6E	00	FD	66	01	DF	28	0E	E7
72D8	01	04	00	FD	09	FD	7D	B7	3C
72E0	20	ED	21	00	00	C9	36	20	4D
72E8	23	36	20	21	00	00	FD	75	0C
72F0	00	FD	74	01	2A	02	42	3A	1A
72F8	00	42	07	3C	5F	16	00	19	13
7300	22	02	42	3A	0B	42	3D	32	5C
7308	0B	42	21	00	00	3E	08	D3	87
7310	F8	3E	0F	D3	F9	C9	00	00	DA
7318	00	00	00	00	00	00	00	00	00
7320	3E	30	3C	09	3B	FC	3D	ED	11
7328	42	12	13	C9	D5	ED	5B	0C	59
7330	42	DF	38	1A	EB	01	D0	07	36
7338	09	22	0C	42	EB	11	E4	79	D2
7340	1A	FE	B0	20	06	13	7B	FE	7A
7348	EE	20	F5	3E	B0	12	D1	01	D5
7350	F0	D8	CD	20	73	01	18	FC	3D
7358	CD	20	73	01	9C	FF	CD	20	E9
7360	73	01	F6	FF	CD	20	73	01	CA
7368	FF	FF	CD	20	73	21	D8	79	D0
7370	11	00	44	01	28	00	ED	B0	1B
7378	C9	00	00	00	00	00	00	00	C9
7380	DD	21	70	78	DD	6E	00	DD	0E
7388	66	01	7C	B5	28	36	7E	FE	72
7390	A0	38	1A	36	20	11	D8	FF	30
7398	19	11	28	44	DF	30	05	21	CB
73A0	00	00	18	14	7E	FE	80	38	60

73A8	07	FE	A0	30	03	CD	98	72	AF
73B0	3A	0A	42	E6	01	C6	A6	77	50
73B8	DD	75	00	DD	74	01	11	00	B5
73C0	AC	19	36	09	DD	23	DD	23	04
73C8	DD	7D	FE	80	C2	84	73	DD	6E
73D0	21	50	78	DD	6E	00	DD	66	77
73D8	01	7C	B5	28	28	DD	7E	02	DF
73E0	07	C6	AA	FE	B0	20	08	36	83
73E8	20	23	36	20	21	00	00	DD	97
73F0	75	00	DD	74	01	77	23	3C	9D
73F8	77	11	FF	AB	19	D6	09	77	A1
7400	23	77	DD	34	02	DD	23	DD	8A
7408	23	DD	23	DD	7D	FE	68	C2	A5
7410	D3	73	2A	02	42	11	DE	79	1C
7418	CD	2C	73	2A	02	42	ED	5B	22
7420	04	42	DF	38	09	22	04	42	CE
7428	11	F8	79	CD	4F	73	C9	00	DD
7430	00	00	00	00	00	00	00	00	00
7438	2A	06	42	06	03	7E	FE	A0	97
7440	38	09	36	20	23	10	F6	2B	EB
7448	2B	2B	C9	F1	C3	30	75	00	78
7450	06	A0	3A	0A	42	E6	01	28	3B
7458	02	06	A3	78	77	23	3C	77	70
7460	23	3C	77	11	FE	AB	19	36	DF
7468	06	23	36	02	23	36	06	C9	89
7470	00	00	00	00	00	00	00	00	00
7478	00	00	00	00	00	00	00	00	00
7480	21	00	44	11	01	44	01	FF	BB
7488	03	36	20	ED	80	21	08	79	68
7490	11	00	44	01	28	00	ED	B0	1B
7498	21	00	78	11	00	F0	01	28	C3
74A0	00	ED	B0	3A	01	42	3C	E6	3C
74A8	07	32	01	42	21	AB	47	22	B1
74B0	06	42	21	00	79	11	80	78	EB
74B8	01	80	00	ED	B0	21	01	00	40
74C0	22	08	42	21	50	78	11	51	B7
74C8	78	01	2F	00	36	00	ED	B0	7B
74D0	3E	20	32	08	42	3A	00	42	59
74D8	3C	32	00	42	21	28	78	11	82
74E0	3C	46	01	10	00	ED	B0	2A	5A
74E8	06	42	CD	50	74	3E	18	32	61
74F0	89	71	06	0C	C5	CD	00	71	0F
74F8	CD	C0	71	21	3C	F2	11	3D	9B
7500	F2	01	0F	00	ED	5F	77	ED	B2
7508	B0	01	00	18	CD	60	00	C1	B7
7510	10	E2	3E	20	32	89	71	21	90
7518	3C	46	11	3D	46	01	0F	00	26
7520	77	ED	B0	C3	00	72	00	00	49
7528	00	00	00	00	00	00	00	00	00
7530	06	1F	3E	07	D3	F8	3E	E7	5A
7538	D3	F9	C5	3E	09	D3	F8	78	1B
7540	E6	0F	D3	F9	3E	06	D3	F8	D0
7548	3E	1F	90	D3	F9	3A	0A	42	3F
7550	3C	32	0A	42	CD	00	71	CD	C5
7558	C0	71	CD	80	73	2A	50	78	E3
7560	7C	B5	20	18	2A	06	42	ED	C8
7568	5F	E6	01	28	01	23	22	50	04
7570	78	32	52	78	3E	08	D3	F8	85
7578	3E	0F	D3	F9	C1	10	BB	21	C6
7580	EE	79	7E	FE	B0	28	09	2B	EF
7588	7D	FE	E3	20	F5	C3	A0	75	4B
7590	36	20	3A	01	42	3D	32	01	43

7598	42	C3	80	74	00	00	00	00	F9
75A0	21	FA	79	11	60	7D	01	06	89
75A8	00	ED	80	C3	00	70	00	00	D0
75B0	21	00	00	22	04	42	31	00	AA
75B8	60	C3	00	70	00	00	00	00	33
75C0	00	00	00	00	00	00	00	00	00
75C8	00	00	00	00	00	00	00	00	00
75D0	00	00	00	00	00	00	00	00	00
75D8	00	00	00	00	00	00	00	00	00
75E0	00	00	00	00	00	00	00	00	00
75E8	00	00	00	00	00	00	00	00	00
75F0	00	00	00	00	00	00	00	00	00
75F8	00	00	00	00	00	00	00	00	00
7600	00	00	00	00	00	00	00	00	00
7608	00	00	00	00	00	00	00	00	00
7610	00	00	00	00	00	00	00	00	00
7618	00	00	00	00	00	00	00	00	00

Nun folgen die Tabellen und Anfangsgrafik:

7800	02	02	02	02	02	03	03	03	13
7808	03	03	03	06	06	06	06	06	27
7810	06	06	06	06	06	06	06	02	2C
7818	02	02	02	02	02	02	02	02	10
7820	02	03	03	03	03	03	03	03	17
7828	2A	2A	20	47	45	54	20	52	C6
7830	45	41	44	59	20	2A	2A	20	B7
7838	00	00	00	00	00	00	00	00	00
7840	00	00	00	00	00	00	00	00	00
7848	00	00	00	00	00	00	00	00	00
7850	00	00	00	00	00	00	00	00	00
7858	00	00	00	00	00	00	00	00	00
7860	00	00	00	00	00	00	00	00	00
7868	00	00	00	00	00	00	00	00	00
7870	00	00	00	00	00	00	00	00	00
7878	00	00	00	00	00	00	00	00	00
7880	00	00	00	00	00	00	00	00	00
7888	00	00	00	00	00	00	00	00	00
7890	00	00	00	00	00	00	00	00	00
7898	00	00	00	00	00	00	00	00	00
78A0	00	00	00	00	00	00	00	00	00
78A8	00	00	00	00	00	00	00	00	00
78B0	00	00	00	00	00	00	00	00	00
78B8	00	00	00	00	00	00	00	00	00
78C0	00	00	00	00	00	00	00	00	00
78C8	00	00	00	00	00	00	00	00	00
78D0	00	00	00	00	00	00	00	00	00
78D8	00	00	00	00	00	00	00	00	00
78E0	00	00	00	00	00	00	00	00	00
78E8	00	00	00	00	00	00	00	00	00
78F0	00	00	00	00	00	00	00	00	00
78F8	00	00	00	00	00	00	00	00	00
7900	50	44	00	00	54	44	00	00	2C
7908	58	44	00	00	5C	44	00	00	3C
7910	60	44	00	00	64	44	00	00	4C
7918	68	44	00	00	6C	44	00	00	5C
7920	A2	44	00	00	A6	44	00	00	D0
7928	AA	44	00	00	AE	44	00	00	E0
7930	B2	44	00	00	B6	44	00	00	F0
7938	BA	44	00	00	F4	44	00	00	36
7940	F8	44	00	00	FC	44	00	00	7C

7948	00	45	00	00	04	45	00	00	8E
7950	08	45	00	00	46	45	00	00	D8
7958	4A	45	00	00	4E	45	00	00	22
7960	52	45	00	00	56	45	00	00	32
7968	98	45	00	00	9C	45	00	00	BE
7970	A0	45	00	00	A4	45	00	00	CE
7978	EA	45	00	00	F2	45	00	00	66
7980	00	00	00	00	00	00	00	00	00
7988	00	00	00	00	00	00	00	00	00
7990	00	00	00	00	00	00	00	00	00
7998	00	00	00	00	00	00	00	00	00
79A0	00	00	00	00	00	00	00	00	00
79A8	00	00	00	00	00	00	00	00	00
79B0	00	00	00	00	00	00	00	00	00
79B8	00	00	00	00	00	00	00	00	00
79C0	00	00	00	00	00	00	00	00	00
79C8	00	00	00	00	00	00	00	00	00
79D0	00	00	00	00	00	00	00	00	00
79D8	B6	B7	B8	B9	BA	20	30	30	18
79E0	32	31	32	20	20	20	20	20	35
79E8	20	20	20	20	20	20	20	B1	91
79F0	B2	B3	B4	B5	B6	B7	B8	B9	AC
79F8	BA	20	20	30	36	37	33	30	FA
7A00	20	20	CA	CA	CA	CA	CA	20	52
7A08	CD	CA	CA	CA	E0	20	CA	20	15
7A10	20	20	20	20	CD	CA	CA	CA	AB
7A18	E0	20	CD	CA	CA	CA	E0	20	2B
7A20	CD	CA	CA	CA	E0	20	20	20	6B
7A28	20	20	20	20	20	CD	E2	20	6F
7A30	CA	20	20	20	CA	20	CA	20	FE
7A38	20	20	20	20	CA	20	20	20	AA
7A40	CA	20	CA	20	20	20	D1	20	05
7A48	CA	20	20	20	CA	20	20	20	54
7A50	20	20	20	20	CD	E2	20	20	6F
7A58	CA	20	20	20	CA	20	CA	20	FE
7A60	20	20	20	20	CA	20	20	20	AA
7A68	CA	20	CA	20	20	20	20	20	54
7A70	CA	20	20	20	CA	20	20	20	54
7A78	20	20	20	CD	E2	20	20	20	6F
7A80	CA	CA	CA	CA	CA	20	CA	20	FC
7A88	20	20	20	20	CA	CA	CA	CA	AB
7A90	CA	20	CA	20	CA	CA	CA	20	52
7A98	CA	CA	CA	CA	CA	20	20	20	52
7AA0	20	20	CD	E2	20	20	20	20	6F
7AA8	CA	20	20	20	CA	20	CA	20	FE
7AB0	20	20	20	20	CA	20	20	20	AA
7AB8	CA	20	CA	20	20	20	CA	20	FE
7AC0	CA	20	20	20	CA	20	20	20	54
7AC8	20	20	CA	20	20	20	20	20	AA
7AD0	CA	20	20	20	CA	20	CA	20	FE
7AD8	20	20	20	20	CA	20	20	20	AA
7AE0	CA	20	CA	20	20	20	CA	20	FE
7AE8	CA	20	20	20	CA	20	20	20	54
7AF0	20	20	CA	CA	CA	CA	CA	20	52
7AF8	CA	20	20	20	CA	20	CA	CA	AB
7B00	CA	CA	CA	20	CA	20	20	20	AB
7B08	CA	20	E4	CA	CA	CA	E2	20	2E
7B10	CA	20	20	20	CA	20	20	20	54
7B18	20	20	20	20	20	20	20	20	00
7B20	20	20	20	20	20	20	20	20	00
7B28	20	20	20	20	20	20	20	20	00

7B30	20	20	20	20	20	20	20	20	00
7B38	20	20	20	20	20	20	20	20	00
7B40	20	20	44	69	65	20	5A	61	2D
7B48	6C	6F	6E	65	6E	20	61	75	12
7B50	73	20	64	65	72	20	47	61	96
7B58	6C	61	78	69	65	20	4E	47	C8
7B60	43	2D	35	34	37	20	20	20	70
7B68	20	20	67	72	65	69	66	65	B2
7B70	6E	20	64	69	65	20	45	72	97
7B78	64	65	20	6D	69	74	20	69	BC
7B80	68	72	65	72	20	67	65	73	10
7B88	61	6D	74	65	6E	20	20	20	75
7B90	20	20	46	6C	6F	74	74	65	AE
7B98	20	61	6E	2E	20	53	69	65	5E
7BA0	20	73	69	6E	64	20	6D	69	C4
7BA8	74	20	64	65	6D	20	6C	65	8B
7BB0	74	7A	74	65	6E	20	20	20	95
7BB8	20	20	46	6C	61	67	67	73	94
7BC0	63	68	69	66	66	20	64	65	E9
7BC8	72	20	52	61	75	6D	2D	42	96
7BD0	61	73	69	73	20	48	61	6E	E7
7BD8	67	65	6C	61	72	20	20	20	68
7BE0	20	20	67	65	66	6C	6F	68	05
7BE8	65	6E	20	75	6E	64	20	6D	C7
7BF0	75	65	73	73	65	6E	20	73	26
7BF8	69	63	68	20	64	75	72	63	02
7C00	68	20	64	69	65	20	20	20	1A
7C08	20	20	20	20	20	66	65	69	D4
7C10	6E	64	6C	69	63	68	65	6E	45
7C18	20	52	65	69	68	65	6E	20	9B
7C20	68	61	65	6D	70	66	65	6E	47
7C28	2E	20	20	20	20	20	20	20	0E
7C30	20	20	20	20	20	20	20	20	00
7C38	20	20	20	20	20	20	20	20	00
7C40	20	20	20	20	20	20	20	20	00
7C48	20	20	20	20	20	20	20	20	00
7C50	20	20	20	20	20	20	20	20	00
7C58	20	20	44	69	65	20	53	74	39
7C60	65	75	65	72	75	6E	67	20	1B
7C68	64	65	73	20	53	63	68	69	E3
7C70	66	66	73	20	67	65	73	63	01
7C78	68	69	65	68	74	20	20	20	72
7C80	20	20	6D	69	74	20	64	65	73
7C88	6E	20	50	66	65	69	6C	74	F2
7C90	61	73	74	65	6E	2E	20	49	B2
7C98	68	72	20	4C	61	73	65	72	F1
7CA0	20	77	69	72	64	20	20	20	36
7CA8	20	20	20	20	20	6D	69	74	EA
7CB0	20	64	65	72	20	4C	65	65	91
7CB8	72	74	61	73	74	65	20	61	14
7CC0	75	73	67	65	6C	6F	65	73	67
7CC8	74	2E	20	20	20	20	20	20	62
7CD0	20	20	20	20	20	20	20	20	00
7CD8	20	20	20	20	20	20	20	20	00
7CE0	20	20	20	20	20	20	20	20	00
7CE8	20	20	20	20	20	20	20	20	00
7CF0	20	20	20	20	20	20	20	20	00
7CF8	20	20	0F	03	03	03	03	03	3E
7D00	03	03	03	03	03	03	03	03	98
7D08	03	03	03	03	03	03	03	03	98
7D10	03	03	03	03	03	03	03	03	98
7D18	03	03	03	03	E3	20	20	20	8F

7D20	20	20	CF	47	45	53	43	48	79
7D28	57	49	4E	44	49	47	48	45	52
7D30	49	54	20	28	30	2E	2E	2E	9F
7D38	39	29	20	20	35	20	20	20	37
7D40	20	20	20	20	DB	20	20	20	8B
7D48	20	20	CF	42	45	53	54	45	82
7D50	53	20	45	52	47	45	42	4E	26
7D58	49	53	20	20	20	20	20	20	5C
7D60	20	30	30	30	30	30	20	20	50
7D68	20	20	20	20	DB	20	20	20	8B
7D70	20	20	CF	44	72	75	65	63	02
7D78	6B	65	6E	20	53	69	65	20	9F
7D80	3C	53	3E	20	75	6D	20	7A	69
7D88	75	20	62	65	67	69	6E	6E	08
7D90	65	6E	20	21	DB	20	20	20	4F
7D98	20	20	E3	DA	DA	DA	DA	DA	65
7DA0	DA	DA	DA	DA	DA	DA	DA	DA	00
7DA8	DA	DA	DA	DA	DA	DA	DA	DA	00
7DB0	0A	DA	DA	DA	DA	DA	DA	DA	00
7DB8	DA	DA	DA	DA	DF	20	20	20	A7
7DC0	20	20	20	20	20	20	20	20	00
7DC8	20	20	20	20	20	20	20	20	00
7DD0	20	20	20	20	20	20	20	20	00
7DD8	20	20	20	20	20	20	20	20	00
7DE0	20	20	20	20	20	20	20	20	00
7DE8	00	00	00	00	00	00	00	00	00
7DF0	00	00	00	00	00	00	00	00	00
7DF8	00	00	00	00	00	00	00	00	00
7E00	02	02	07	0E	1C	3F	0C	18	98
7E08	40	40	E0	70	38	FC	30	18	4C
7E10	04	02	07	0E	1C	3F	0C	06	88
7E18	20	40	E0	70	38	FC	30	60	74
7E20	0F	19	39	1F	0F	0A	14	0A	B7
7E28	F0	98	9C	F8	F0	50	28	50	D4
7E30	0F	19	39	1F	0F	0A	14	28	D5
7E38	F0	98	9C	F8	F0	50	28	14	98
7E40	07	1F	29	29	1F	0F	03	01	AA
7E48	E0	F8	94	94	F8	F0	C0	80	28
7E50	07	1F	32	32	1F	0F	03	01	BC
7E58	E0	F8	4C	4C	F8	F0	C0	80	98
7E60	01	02	04	F9	7F	2B	01	00	AB
7E68	80	40	20	9F	FE	D4	80	00	D1
7E70	01	03	04	09	3F	6B	A1	00	5C
7E78	80	C0	20	90	FC	D6	85	00	47
7E80	00	82	42	31	2E	4A	11	00	7E
7E88	00	41	42	8C	74	52	88	00	5D
7E90	00	02	02	01	FE	2A	09	00	36
7E98	00	40	40	80	7F	54	90	00	63
7EA0	00	01	05	04	08	08	05	05	24
7EA8	50	50	08	08	10	50	40	00	50
7EB0	05	05	08	08	04	05	01	00	24
7EB8	00	40	50	10	08	08	50	50	50
7EC0	00	0D	13	23	45	48	30	00	00
7EC8	00	B0	C8	C4	A2	12	0C	00	FC
7ED0	00	7D	83	83	7D	00	00	00	00
7ED8	00	BE	C1	C1	BE	00	00	00	FE
7EE0	00	3C	24	27	27	24	3C	00	0E
7EE8	00	3C	24	E4	E4	24	3C	00	88
7EF0	00	0F	09	39	39	09	0F	00	A2
7EF8	00	F0	90	9C	9C	90	F0	00	38
7F00	00	04	0E	1B	1B	0E	04	00	5A

7F08	18	18	3C	DB	99	7E	7E	24	00
7F10	00	20	70	D8	D8	70	20	00	D0
7F18	00	04	0E	15	15	0E	04	00	4E
7F20	18	18	3C	E7	C3	7E	7E	24	36
7F28	00	20	70	A8	A8	70	20	00	70
7F30	10	08	10	08	10	00	00	00	40
7F38	00	00	00	08	10	08	10	08	38
7F40	18	18	18	18	00	00	00	00	60
7F48	00	00	00	00	18	18	18	18	60
7F50	11	0C	03	02	0F	10	01	06	48
7F58	08	B0	40	FC	B2	E0	18	04	A2
7F60	01	14	01	55	02	24	01	08	9A
7F68	10	80	24	40	AA	80	28	80	C6
7F70	00	00	04	00	12	00	04	00	1A
7F78	00	90	00	40	00	08	40	00	18
7F80	10	10	D6	FE	D6	10	00	00	DA
7F88	04	04	4C	54	64	44	40	40	D0
7F90	08	10	30	10	10	18	10	20	B0
7F98	18	20	40	48	58	48	50	20	D0
7FA0	04	04	4C	54	64	44	40	40	D0
7FA8	00	00	00	08	10	20	00	00	38
7FB0	18	20	40	58	34	04	08	30	40
7FB8	18	20	40	40	40	44	48	30	B4
7FC0	18	24	44	44	44	44	48	30	C4
7FC8	18	24	44	48	50	68	44	40	04
7FD0	0C	10	20	28	30	24	28	30	10
7FD8	FF	00	E3	14	14	14	14	14	46
7FE0	C7	08	93	24	48	4F	40	3F	9C
7FEB	F3	04	E4	02	01	BF	00	BF	5C
7FF0	FC	00	F8	40	20	A0	20	C0	04
7FF8	00	00	00	00	00	00	00	00	00

7000	21007E	LD	HL, 7E00H	; Zeichenatz von 7E00H
7003	1100F4	LD	DE, 0F400H	; nach F400H in den Zeichenspeicher
7006	010002	LD	BC, 0200H	; verschieben.
7009	EDB0	LDIR		;
700B	3E01	LD	A, 01H	; CHAR 2 einschalten. Argument
700D	CDB03F	CALL	3FBOH	; in Akku ist um eins vermindert.
7010	CDFO70	CALL	70FOH	; Unterprg.: Soundoff und LGR
7013	21007A	LD	HL, 7A00H	; Adresse der Anfangsgrafik
7016	110044	LD	DE, 4400H	; Adresse Bildschirmumfang
7019	010004	LD	BC, 0400H	; Anzahl der Bytes
701C	EDB0	LDIR		; Verschieben.
701E	2100F0	LD	HL, 0F000H	; Farbspeicher
7021	1101F0	LD	DE, 0F001H	; Farbspeicher +1
7024	01FF03	LD	BC, 03FFH	; Laenge -1
7027	3606	LD	(HL), 06H	; erstes Byte mit 06H belegen
7029	EDB0	LDIR		; 1. Byte auf 2. usw. schieben.
702B	CDFO14	CALL	14FOH	; Unterprg.: Zufallszahl erzeugen.
702E	2AAA40	LD	HL, (40AAH)	; Zahl in HL holen.
7031	7C	LD	A, H	; MSB in Akku
7032	E601	AND	01H	; Akku wird 00H oder 01H
7034	C644	ADD	A, 44H	; Akku wird 44H oder 45H
7036	67	LD	H, A	; Zurück in MSB. (HL = 4400H - 45FFH)
7037	FE44	CP	44H	; Wenn MSB = 44H,
7039	2B05	JR	Z, S+07H	; dann nach 7040H
703B	7D	LD	A, L	; LSB in Akku
703C	FE1B	CP	1BH	; Wenn >= 1BH
703E	30EB	JR	NC, S-13H	; Dann neue Zahl erzeugen.
7040	7E	LD	A, (HL)	; HL Zeiger auf Bildschirmadresse
7041	FE7F	CP	7FH	; Zeichen auf HL = 7FH ?
7043	2B0A	JR	Z, S+0CH	; Wenn ja dann nach 704FH
7045	367F	LD	(HL), 7FH	; sonst 7FH auf HL abspeichern
7047	1100AC	LD	DE, 0AC00H	; Offset zum Farbspeicher
704A	19	ADD	HL, DE	; zu HL addieren
704B	3602	LD	(HL), 02H	; 02H auf HL abspeichern (rot)
704D	1B10	JR	S+12H	; Weiter bei 705FH
704F	E5	PUSH	HL	; HL auf den Stack
7050	D1	POP	DE	; DE zurück
7051	EB	EX	DE, HL	; vertauschen
7052	010036	LD	BC, 3600H	; Offset zur Anfangsgrafik
7055	09	ADD	HL, BC	; zu HL addieren
7056	7E	LD	A, (HL)	; Originalzeichen der Anfangsgrafik
7057	12	LD	(DE), A	; in den Bildschirmspeicher speichern
7058	EB	EX	DE, HL	; Bildschirmadresse zurück nach HL
7059	1100AC	LD	DE, 0AC00H	; Offset zum Farbspeicher
705C	19	ADD	HL, DE	; zu HL addieren
705D	3605	LD	(HL), 05H	; 02H auf HL abspeichern (gruen)
705F	CD2B00	CALL	002BH	; Unterprg.: GETCHR (Tastatureabfrage)
7062	FE30	CP	30H	; (' 30H ASC("0")
7064	DA2B70	JP	C, 702BH	; Wenn ja, dann Sprung zur Schleife
7067	FE3A	CP	3AH	; (' 3AH ASC("A")
7069	3B08	JR	C, S+0AH	; Wenn ja, dann zu 7073H
706B	FE53	CP	53H	; (' 53H ASC("S")
706D	C22B70	JP	NZ, 702BH	; Wenn nicht, dann zur Schleife
7070	C3B070	JP	70B0H	; Sonst Spiel starten
7073	323C47	LD	(473CH), A	; gedruckte Zahl in Bildschirm anzeigen
7076	D630	SUB	30H	; - 30H ergibt einen Wert von 00H-09H
7078	320042	LD	(4200H), A	; in Variable fuer Level abspeichern
707B	C32B70	JP	702BH	; und zurück zur Schleife.

; *** STARTEN DES SPIELS ***

7080	210000	LD	HL, 0000H	; HL 0000 mit Startpunktzahl belegen
7083	220242	LD	(4202H), HL	; in Variable fuer Punktzahl abspeichern
7086	3E07	LD	A, 07H	; Zuehler fuer Feindtyp auf 7 setzen
7088	320142	LD	(4201H), A	; und abspeichern.
708B	3A3C47	LD	A, (473CH)	; Level aus Bildechirmpeicher holen
708E	D630	SUB	30H	; - 30H
7090	320042	LD	(4200H), A	; in Variable fuer Level abspeichern
7093	3E07	LD	A, 07H	; 07H (SOUND 7, 247 :
7095	D3F8	OUT	(0F8H), A	; auf Port F8H schaltet Kauschen
7097	3EF7	LD	A, 0F7H	; 77H auf Kanal 1 ein)
7099	D3F9	OUT	(0F9H), A	; auf Port F9H
709B	21E479	LD	HL, 79E4H	; Auf die Adresse fuer die Anzeige der
709E	0605	LD	B, 05H	; restlichen Schiffe, 5 mal
70A0	36B0	LD	(HL), 0B0H	; 0B0H (kleines Schiff) abspeichern
70A2	23	INC	HL	;
70A3	10FB	DJNZ	S-03H	;
70A5	3E05	LD	A, 05H	; 05H
70A7	D3F8	OUT	(0F8H), A	; auf Port F8H (SOUND 5, 3 :
70A9	3E03	LD	A, 03H	; 03H Frequenz des dritten
70AB	D3F9	OUT	(0F9H), A	; auf Port F9H Kanale vorbelegen)
70AD	21D007	LD	HL, 07D0H	; 07D0H = 2000D als erste Grenze fuer
70B0	220C42	LD	(420CH), HL	; Bonusschiff abspeichern
70B3	C38074	JP	7480H	; Sprung zum Bildaufbau

; *** UNTERPROGRAMM SOUND0FF UND LGR ***

70F0	3E07	LD	A, 07H	; 07H (SOUND 7, 255 :
70F2	D3F8	OUT	(0F8H), A	; auf Port F8H alle Kanäle
70F4	3EFF	LD	A, 0FFH	; 0FFH abschalten)
70F6	D3F9	OUT	(0F9H), A	; auf Port F9H
70F8	CDB038	CALL	38B0H	; Ram-Routine fuer LGR
70FB	C9	RET		;

; *** UNTERPROGRAMM FUER BEWEGUNG DER FEINDE ***

7100	DD218078	LD	IX, 78B0H	; IX auf Tabelle der Feindpositionen
7104	ED4B0842	LD	BC, (4208H)	; BC mit X-Bewegungsrichtung vorbelegen
7108	DD6E00	LD	L, (IX+00H)	; L ist LSB der Position
710B	DD6601	LD	H, (IX+01H)	; H ist MSB der
710E	7C	LD	A, H	; MSB
710F	B5	OR	L	; oderiert mit LSB = 0 ?
7110	CAB371	JP	Z, 71B3H	; Wenn ja, dann existiert Feind nicht
7113	3620	LD	(HL), 20H	; Erstes Byte mit 20H (SPACE) loeschen
7115	23	INC	HL	; und
7116	3620	LD	(HL), 20H	; zweites Byte ebenfalls

7118	2B	DEC	HL	; HL zurueck auf erstes Byte
7119	09	ADD	HL, BC	; X-Richtung addieren
711A	005E02	LD	E, (IX+02H)	; E ist LSB der Y-Richtung
711D	1600	LD	D, 00H	; D loeschen
711F	7B	LD	A, E	; LSB in Akku
7120	FED8	CP	008H	; = 008H ? (= -40D)
7122	2002	JR	NZ, \$+04H	; nein, es ist 00H oder 28H (= 40D)
7124	16FF	LD	D, 0FFH	; sonst MSB = 0FFH fuer Addition
7126	19	ADD	HL, DE	; Y-Richtung zur Position addieren
7127	87	OR	A	; Y-Richtung + 0 ?
7128	280C	JR	Z, \$+0EH	; wenn ja, dann nach 7136H
712A	FE28	CP	28H	; = +40D ?
712C	2805	JR	Z, \$+07H	; wenn ja, dann nach 7133H
712E	DD3503	DEC	(IX+03H)	; sonst Y-Bewegungszaeher -1
7131	1803	JR	\$+05H	; und nach 7136H
7133	DD3403	INC	(IX+03H)	; Y-Bewegungszaeher +1
7136	DD7E03	LD	A, (IX+03H)	; Y-Bewegungszaeher in Akku
7139	B7	OR	A	; Flags setzen
713A	2004	JR	NZ, \$+06H	; () 0 ? dann nach 7140H
713C	DD360200	LD	(IX+02H), 00H	; sonst LSB der Y-Richtung loeschen
7140	FE18	CP	18H	; = 18H
7142	2008	JR	NZ, \$+0AH	; wenn nicht, dann nach 714CH
7144	DD360200	LD	(IX+02H), 00H	; sonst LSB der Y-Richtung
7148	DD360300	LD	(IX+03H), 00H	; und Y-Bewegungszaeher loeschen
714C	11E847	LD	DE, 47E8H	; Adresse von Bildschirmende +1
714F	DF	RST	18H	; mit HL vergleichen (Position)
7150	3804	JR	C, \$+06H	; wenn HL < DE dann nach 7156H
7152	1140FC	LD	DE, 0FC40H (= -960D)	; sonst 0FC40H (= -960D) zur Position
7155	19	ADD	HL, DE	; addieren. (HL= zweite Bildschirmzeile
7156	112844	LD	DE, 4428H	; Adresse der 2. Bildschirmzeile in DE
7159	DF	RST	18H	; mit HL vergleichen
715A	3004	JR	NC, \$+06H	; wenn HL > DE dann nach 7160H
715C	11C003	LD	DE, 03C0H	; sonst 03C0H (= 960D) zur Position
715F	19	ADD	HL, DE	; addieren. (HL= letzte Bildschirmzeile)
7160	DD7500	LD	(IX+00H), L	; LSB der Position zurueck in Tabelle
7163	DD7401	LD	(IX+01H), H	; MSB ebenso
7166	3A0142	LD	A, (4201H)	; Feindtyp in Akku
7169	E607	AND	07H	; nur von 00H bis 07H zugelassen
716B	07	RLCA		; 2 mal links rotieren
716C	07	RLCA		; d.h. mit 4 multiplizieren
716D	C6B0	ADD	A, 80H	; + 80H ergibt erstes Zeichen des Feinds
716F	5F	LD	E, A	; in E zwischenspeichern
7170	3A0A42	LD	A, (420AH)	; Zaehler in den Akku
7173	E602	AND	02H	; mit 02H undieren. (gibt 00H oder 02H)
7175	83	ADD	A, E	; E addieren, ergibt Phase des Feinds
7176	77	LD	(HL), A	; auf Position abspeichern
7177	23	INC	HL	; naechste Position
7178	3C	INC	A	; naechstes Zeichen
7179	77	LD	(HL), A	; abspeichern. Feind steht in Bildschirm
717A	11FFAB	LD	DE, 0ABFFH	; Offset zum Farbspeicher -1 (wegen INC)
717D	19	ADD	HL, DE	; zur Position addieren
717E	3A0142	LD	A, (4201H)	; Feindtyp
7181	3C	INC	A	; +1 als Farbcode
7182	77	LD	(HL), A	; in den Farbspeicher ablegen
7183	23	INC	HL	; und
7184	77	LD	(HL), A	; auf der naechsten Adresse
7185	DD7E03	LD	A, (IX+03H)	; Y-Bewegungszaeher in Akku
7188	B7	OR	A	; Flags setzen
7189	2013	JR	NZ, \$+15H	; () 0 ? dann nach 7192H
718B	DDF014	CALL	14F0H	; Zufallszahl erzeugen
718E	3A0042	LD	A, (4200H)	; Level in Akku
7191	5F	LD	E, A	; in E zwischenspeichern

7192 3AAA40 LD A,(40AAH) ; Zufallszahl in Akku
 7195 BB CP E ; mit Level vergleichen
 7196 3004 JR NC,\$+06H ; wenn Zahl > Level dann nach 719CH
 7198 DD360228 LD (IX+02H),28H ; 28H (= +40D) in Y-Richtung abspeichern
 719C 1815 JR \$+17H ; und nach 7183H
 719E DD7E02 LD A,(IX+02H) ; Y-Richtung in den Akku
 71A1 FE28 CP 28H ; = 28H (= +40D)
 71A3 200E JR NZ,\$+10H ; wenn nicht, dann nach 7183H
 71A5 CDF014 CALL 14F0H ; Zufallszahl erzeugen
 71A8 3AAA40 LD A,(40AAH) ; Zahl in Akku
 71AB FE01 CP 01H ; (01H (können Sie ändern)
 71AD 3004 JR NC,\$+06H ; wenn nicht, dann nach 7183H
 71AF DD3602D8 LD (IX+02H),0D8H ; sonst 0D8H (= -40D) in Y-Richtung
 71B3 110400 LD DE,0004H ; 0004H fuer 4 Tabellenplätze
 71B6 DD19 ADD IX,DE ; zu IX addieren
 71B8 DD DEFB 0DDH ; DD7D ist ein Sonderbefehl, der das
 71B9 7D LD A,L ; LSB von IX in den Akku lädt.
 71BA B7 OR A ; gleich 0 ? (bei IX = 7900H)
 71BB C20471 JP NZ,7104H ; nein ? dann nächsten Feind bewegen
 71BE C9 RET ; sonst zurück

; *** UNTERPROGRAMM ZUR ÄNDERUNG DER X-RICHTUNG ***

71C0 3E08 LD A,08H ; 08H
 71C2 D3F8 OUT (0F8H),A ; auf Port F8H
 71C4 DBF9 IN A,(0F9H) ; Port F9H in den Akku (Leutafuerke I)
 71C6 B7 OR A ; Flags setzen
 71C7 2801 JR Z,\$+03H ; schon 0 ? dann nach 71CAH
 71C9 3D DEC A ; sonst -1
 71CA D3F9 OUT (0F9H),A ; und auf Port F9H wieder ausgeben.
 71CC 212844 LD HL,4428H ; HL auf 1. Byte, 2. Bildechirnzeile
 71CF 112800 LD DE,0028H ; DE = 28H (= +40D)
 71D2 0616 LD B,16H ; B = 16H (= 20D) als Zaehler der Zeilen
 71D4 7E LD A,(HL) ; Zeichen aus dem Bildechirm in Akku
 71D5 FE20 CP 20H ; = 20H (SPACE) ?
 71D7 2005 JR NZ,\$+07H ; wenn nicht, dann nach 71DEH
 71D9 19 ADD HL,DE ; sonst HL = HL + 40D (nächste Zeile)
 71DA 10F8 DJNZ \$-06H ; Zaehler -1 nicht 0 ? dann nach 71D4H
 71DC 1806 JR \$+08H ; sonst nach 71E4H
 71DE 210100 LD HL,0001H ; +1 (= nach rechts)
 71E1 220842 LD (4208H),HL ; in Variable fuer X-Richtung
 71E4 214F44 LD HL,444FH ; HL auf 39. Byte in 2. Bildechirnzeile
 71E7 0616 LD B,16H ; B = 20D
 71E9 7E LD A,(HL) ; Zeichen in Akku
 71EA FE20 CP 20H ; = 20H (SPACE) ?
 71EC 2004 JR NZ,\$+06H ; wenn nicht, dann nach 71F2H
 71EE 19 ADD HL,DE ; HL = HL + 40D
 71EF 10F8 DJNZ \$-06H ; nächstes Zeichen testen
 71F1 C9 RET ; und zurück
 71F2 21FFFF LD HL,0FFFFH ; -1 (= nach links)
 71F5 220842 LD (4208H),HL ; in Variable fuer X-Richtung
 71F8 C9 RET ; und zurück

*** HAUPTSCHLEIFE ***

7200	3A0A42	LD	A, (420AH)	; Zähler in Akku
7203	3C	INC	A	; +1
7204	320A42	LD	(420AH), A	; wieder abspeichern
7207	CD0071	CALL	7100H	; Unterprg.: Feindbewegung
720A	CD0071	CALL	7100H	; Unterprg.: X-Richtung
720D	CD3874	CALL	7438H	; Unterprg.: Schiff löschen
7210	3A40F8	LD	A, (0F840H)	; Testaturspeicher in Akku
7213	E620	AND	20H	; Bit 5 isolieren (Pfeil nach links)
7215	2801	JR	Z, \$+03H	; nicht gesetzt ? dann nach 7218H
7217	2B	DEC	HL	; sonst Schiffe-Position -1
7218	3A40F8	LD	A, (0F840H)	; Testaturspeicher in Akku
721B	E640	AND	40H	; Bit 6 (Pfeil) nach rechts
721D	2801	JR	Z, \$+03H	; nicht gesetzt ? dann nach 7220H
721F	23	INC	HL	; sonst Schiffe-Position +1
7220	7D	LD	A, L	; LSB der Position in Akku
7221	FE98	CP	98H	; = 98H ? (linker Rand)
7223	2001	JR	NZ, \$+03H	; wenn nicht, nach 7226H
7225	23	INC	HL	; sonst Position +1
7226	FE8C	CP	08CH	; = 8CH ? (rechter Rand -4)
7228	2001	JR	NZ, \$+03H	; wenn nicht, nach 7228H
722A	2B	DEC	HL	; sonst Position -1
722B	220642	LD	(4206H), HL	; Position wieder abspeichern
722E	CD5074	CALL	7450H	; Unterprg.: Schiff und Farbe setzen
7231	3A40F8	LD	A, (0F840H)	; Testaturspeicher in Akku
7234	E680	AND	80H	; Bit 7
7236	2803	JR	Z, \$+05H	; nicht gesetzt ? dann nach 723BH
7238	CD7072	CALL	7270H	; Unterprg.: Schuss initialisieren
723B	CD8073	CALL	7380H	; Unterprg.: Schüsse und Explosionen
723E	3A40F8	LD	A, (0F840H)	; Testaturspeicher in Akku
7241	FE06	CP	06H	; = 06H (BREAK & CLEAR) ?
7243	CA0070	JP	Z, 7000H	; wenn ja, zum Programmstart
7246	3A0B42	LD	A, (420BH)	; Anzahl der verbliebenen Feinde
7249	B7	OR	A	; Flags setzen
724A	C20072	JP	NZ, 7200H	; wenn != 0 dann zur Hauptschleife
724D	C38074	JP	7480H	; sonst nächster Level

*** SCHUSS INITIALISIEREN ***

7270	217078	LD	HL, 7870H	; Anfang der Schusspositionstabelle
7273	7E	LD	A, (HL)	; LSB in Akku
7274	23	INC	HL	; +1
7275	B6	OR	(HL)	; oderiert mit MSB
7276	2807	JR	Z, \$+09H	; = 0 ? dann nach 727PH
7278	23	INC	HL	; nächste Tabellenposition
7279	7D	LD	A, L	; LSB des Zeigers in Akku
727A	FE80	CP	80H	; = 80H ? (HL = 7880H)
727C	20F5	JR	NZ, \$-09H	; wenn nicht, dann nach 7273H

727E	C9	RET		; zurueck, alle Schuesse belegt
727F	2B	DEC	HL	; HL wieder auf LSB
7280	EB	EX	DE, HL	; in DE
7281	2A0642	LD	HL, (4206H)	; Schiffsposition
7284	01D9FF	LD	BC, 0FFD9H	; FFD9H (= -39D)
7287	09	ADD	HL, BC	; addieren, Position ueber dem Schiff
7288	EB	EX	DE, HL	; wieder vertauschen
7289	73	LD	(HL), E	; LSB in Tabelle
728A	23	INC	HL	; naechste Adresse
728B	72	LD	(HL), D	; MSB abspeichern
728C	EB	EX	DE, HL	; Position wieder in HL
728D	36A6	LD	(HL), 0A6H	; AGH ist def. Zeichen fuer Schuss
728F	1100AC	LD	DE, 0AC00H	; Offset zum Farbspeicher
7292	19	ADD	HL, DE	; addieren
7293	3603	LD	(HL), 03H	; und mit 3 beladen
7295	C9	RET		; zurueck

; *** EXPLOSION INITIALISIEREN UND FEIND LOESCHEN ***

7298	E5	PUSH	HL	; Trefferposition retten
7299	FD215078	LD	IY, 7850H	; IY auf Tabelle der Explosionen
729D	FD6E00	LD	L, (IY+00H)	; LSB in L
72A0	FD6601	LD	H, (IY+01H)	; MSB in H
72A3	7C	LD	A, H	; MSB
72A4	B5	OR	L	; oderiert mit LSB
72A5	2812	JR	Z, S+14H	; = 0 ? dann nach 72B9H
72A7	FD23	INC	IY	; sonst drei Tabellenplaetze weiter
72A9	FD23	INC	IY	
72AB	FD23	INC	IY	
72AD	FD	DEFB	0FDH	; Sonderbefehl 'LD A, LSB IY'
72AE	7D	LD	A, L	
72AF	FE68	CP	68H	; = 68H (IY = 7868H) ?
72B1	C29D72	JP	NZ, 729DH	; wenn nicht, dann nach 729DH
72B4	E1	POP	HL	; Stack korrigieren
72B5	210000	LD	HL, 0000H	; naechste Schussposition = 0
72B8	C9	RET		; und zurueck
72B9	E1	POP	HL	; Trefferposition zurueck in HL
72BA	7E	LD	A, (HL)	; getroffenes Zeichen in Akku
72BB	E601	AND	01H	; mit 01H undieren. Bei Ergebnis 0 ist
72BD	2801	JR	Z, S+03H	; es das linke Zeichen eines Feindes
72BF	2B	DEC	HL	; sonst HL decrementieren
72C0	FD7500	LD	(IY+00H), L	; LSB der Position in Tabelle ablegen
72C3	FD7401	LD	(IY+01H), H	; MSB ebenso.
72C6	FD360200	LD	(IY+02H), 00H	; Explosionszaehler auf 0 setzen
72CA	EB	EX	DE, HL	; Position in DE zwischenspeichern
72CB	FD218078	LD	IY, 7880H	; IY auf Positionstabelle der Feinde
72CF	FD6E00	LD	L, (IY+00H)	; LSB der Position in L
72D2	FD6601	LD	H, (IY+01H)	; MSB in H
72D5	DF	RST	18H	; vergleich mit Trefferposition in DE
72D6	280E	JR	Z, S+10H	; gleich ? dann nach 72E6H
72D8	010400	LD	BC, 0004H	; sonst IY +4
72DB	FD09	ADD	IY, BC	; addieren
72DD	FD	DEFB	0FDH	; LSB von IY in Akku
72DE	7D	LD	A, L	
72DF	B7	OR	A	; Flags setzen
72E0	20ED	JR	NZ, S-11H	; () 0 ? dann nach 72CFH

72E2	210000	LD	HL, 0000H	; Naechste Schussposition gleich 0
72E5	C9	RET		; zurueck, kein Feind getroffen
72E6	3620	LD	(HL), 20H	; Feind in Bildschirm loeschen
72E8	23	INC	HL	;
72E9	3620	LD	(HL), 20H	; zweites Byte loeschen
72EB	210000	LD	HL, 0000H	; 0 in Positionstabelle der Feinde
72EE	FD7500	LD	(IY+00H), L	; schreiben. Der entsprechende Feind
72F1	FD7401	LD	(IY+01H), H	; ist geloescht
72F4	2A0242	LD	HL, (4202H)	; Punktzahl in HL
72F7	3A0042	LD	A, (4200H)	; Level in Akku
72FA	07	RLCA		; #2
72FB	3C	INC	A	; +1
72FC	5F	LD	E, A	; nach E speichern
72FD	1600	LD	D, 00H	; D loeschen
72FF	19	ADD	HL, DE	; DE zur Punktzahl addieren
7300	220242	LD	(4202H), HL	; und wieder abspeichern
7303	3A0B42	LD	A, (420BH)	; Anzahl der verbliebenen Feinde in Akku
7306	3D	DEC	A	; -1
7307	320B42	LD	(420BH), A	; wieder abspeichern
730A	210000	LD	HL, 0000H	; naechste Schussposition gleich 0
730D	3E08	LD	A, 08H	; 08H (SOUND 8,15:
730F	D3F8	OUT	(OF8H), A	; auf Port F8H Lautstaerke von Kanal
7311	3E0F	LD	A, 0FH	; 0FH auf maximum setzen
7313	D3F9	OUT	(OF9H), A	; auf Port F9H
7315	C9	RET		; und zurueck

; *** AUSGABE EINER ZAHL ***

7320	3E30	LD	A, 30H	; 30H = ASC('0') in Akku
7322	3C	INC	A	; +1
7323	09	ADD	HL, BC	; HL = HL + BC (BC = -10000, -1000 usw.)
7324	39FC	JR	C, \$-02H	; wenn Carry gesetzt dann weiter
7326	3D	DEC	A	; sonst Akku -1
7327	ED42	SBC	HL, BC	; HL = HL - BC
7329	12	LD	(DE), A	; Akku auf Adresse (DE) abspeichern
732A	13	INC	DE	; DE auf naechste Position
732B	C9	RET		; zurueck
732C	D5	PUSH	DE	; Adresse fuer Ausgabe der Zahl retten
732D	ED5B0C42	LD	DE, (420CH)	; Punktzahl fuer Bonus Schiff nach DE
7331	DF	AST	18H	; mit Punktzahl vergleichen
7332	381A	JR	C, \$+1CH	; Wenn groesser, dann nach 734EH
7334	EB	EX	DE, HL	; Bonuspunktzahl nach HL
7335	01D007	LD	BC, 07D0H	; 07D0H = 2000D
7338	09	ADD	HL, BC	; addieren
7339	220C42	LD	(420CH), HL	; und abspeichern
733C	EB	EX	DE, HL	; Punktzahl zurueck nach HL
733D	11E479	LD	DE, 79E4H	; DE auf Position der Schiffszuige
7340	1A	LD	A, (DE)	; Zeichen in Akku
7341	FEB0	CP	OBOH	; Zeichen fuer Schiff ?
7343	2006	JR	NZ, \$+08H	; wenn nicht, dann nach 734EH
7345	13	INC	DE	; naechste Position
7346	7B	LD	A, E	; LSB in Akku
7347	FEEE	CP	OEEH	; = EEH ? (Hier beginnt 'HIGH-SCORE...')
7349	20F5	JR	NZ, \$-09H	; wenn nicht, dann nach 7340H
734B	3EB0	LD	A, OBOH	; Zeichen fuer Schiff in Akku
734D	12	LD	(DE), A	; auf DE abspeichern

734E	D1	POP	DE	; Adresse fuer Ausgabe zurueck
734F	01F0D8	LD	BC, 0D8F0H	; BC = -10000
7352	CD2073	CALL	7320H	; Stelle berechnen
7355	0118FC	LD	BC, 0FC18H	; BC = -1000
7358	CD2073	CALL	7320H	; Stelle berechnen
735B	019CFF	LD	BC, 0FF9CH	; BC = -100
735E	CD2073	CALL	7320H	; Stelle berechnen
7361	01F6FF	LD	BC, 0FFF6H	; BC = -10
7364	CD2073	CALL	7320H	; Stelle berechnen
7367	01FFFF	LD	BC, 0FFFFH	; BC = -1
736A	CD2073	CALL	7320H	; Stelle berechnen
736D	210879	LD	HL, 7908H	; Adresse der Ausgabeseile
7370	110044	LD	DE, 4400H	; erste Bildschirmzeile
7373	012800	LD	BC, 0028H	; 40 Zeichen
7376	EDB0	LDIR		; kopieren
7378	C9	RST		; und zurueck

; *** UNTERPROGRAMM SCHUESSE UND EXPLOSIONEN BEWEGEN ***

7380	DD217078	LD	IX, 7870H	; IX auf Tabelle der Schuesse
7384	DD6E00	LD	L, (IX+00H)	; LSB der Position nach L
7387	DD6601	LD	H, (IX+01H)	; MSB nach H
738A	7C	LD	A, H	; MSB
738B	B5	OR	L	; oderiert mit LSB
738C	2836	JR	Z, \$+38H	; = 0 ? dann nach 73C4H
738E	7E	LD	A, (HL)	; Zeichen auf Schussposition
738F	FEA0	CP	0A0H	; (A0H ?
7391	381A	JR	C, \$+1CH	; wenn ja, dann nach 73ADH
7393	3620	LD	(HL), 20H	; Schuss loeschen
7395	1108FF	LD	DE, 0FFD8H	; -40D
7398	19	ADD	HL, DE	; zur Position addieren
7399	112844	LD	DE, 4428H	; 2. Bildschirmzeile
739C	DF	RST	18H	; mit Position vergleichen
739D	3005	JR	NC, \$+07H	; wenn groesser dann nach 73A4H
739F	210000	LD	HL, 0000H	; sonst naechste Schussposition gleich 0
73A2	1814	JR	\$+16H	; und nach 73B8H
73A4	7E	LD	A, (HL)	; Zeichen auf neuer Position in Akku
73A5	FE80	CP	80H	; (80H ? (min. der Zeichen fuer Feinde)
73A7	3807	JR	C, \$+09H	; wenn ja, dann nach 73B0H
73A9	FEA0	CP	0A0H	; >= A0H ? (max.)
73AB	3003	JR	NC, \$+05H	; wenn nicht, dann nach 73B0H
73AD	CD9872	CALL	729BH	; Unterprg.: Explosion initialisieren
73B0	3A0A42	LD	A, (420AH)	; Zaehler in Akku
73B3	E601	AND	01H	; mit 01H undieren (ergibt 0 oder 1)
73B5	C6A6	ADD	A, 0A6H	; + A6H (ergibt Zeichen fuer Schuss)
73B7	77	LD	(HL), A	; auf neuer Position abspeichern
73B8	DD7500	LD	(IX+00H), L	; LSB Schussposition wieder in Tabelle
73BB	DD7401	LD	(IX+01H), H	; MSB ebenso
73BE	1100AC	LD	DE, 0AC00H	; Offset zum Farbspeicher
73C1	19	ADD	HL, DE	; addieren
73C2	3609	LD	(HL), 09H	; und mit 09H belegen
73C4	DD23	INC	IX	; Tabellenseiger +2
73C6	DD23	INC	IX	;
73C8	DD	DEFB	0DDH	; LSB von IX in Akku
73C9	7D	LD	A, L	;
73CA	FE80	CP	80H	; = 80H (IX = 7880H)

73CC	C28473	JP	NZ, 7384H	; wenn nicht, nach 7384H
73CF	DD215078	LD	IX, 7850H	; IX auf Tabelle der Explosionen
73D3	DD6E00	LD	L, (IX+00H)	; LSB der Position in L
73D6	DD6601	LD	H, (IX+01H)	; MSB in H
73D9	7C	LD	A, H	; MSB
73DA	B5	OR	L	; oderiert mit LSB
73DB	2828	JR	Z, \$+2AH	; = 0 ? dann nach 7405H
73DD	DD7E02	LD	A, (IX+02H)	; Explosionszähler in Akku
73E0	07	RLCA		; *2
73E1	C6AA	ADD	A, 0AAH	; + AAH (Zeichen der Explosionen)
73E3	FEB0	CP	0B0H	; = B0H ? (Explosion zuende)
73E5	2008	JR	NZ, \$+0AH	; wenn nicht, nach 73E7H
73E7	3620	LD	(HL), 20H	; Explosion löschen
73E9	23	INC	HL	;
73EA	3620	LD	(HL), 20H	; zweites Byte löschen
73EC	210000	LD	HL, 0000H	; Position gleich 0
73EF	DD7500	LD	(IX+00H), L	; wieder abspeichern
73F2	DD7401	LD	(IX+01H), H	;
73F5	77	LD	(HL), A	; Zeichen auf Explosion abspeichern
73F6	23	INC	HL	; nächste Adresse
73F7	3C	INC	A	; nächstes Zeichen
73F8	77	LD	(HL), A	; abspeichern
73F9	11FFAB	LD	DE, 0ABFFH	; Offset zum Farbspeicher -1 (wegen INC)
73FC	19	ADD	HL, DE	; zur Position addieren
73FD	D609	SUB	09H	; Zeichenwert -9 ergibt Farbe (2,4od. 6)
73FF	77	LD	(HL), A	; in den Farbspeicher ablegen
7400	23	INC	HL	; und
7401	77	LD	(HL), A	; auf die nächste Adresse
7402	DD3402	INC	(IX+02H)	; Explosionszähler +1
7405	DD23	INC	IX	; Tabellenzähler +3
7407	DD23	INC	IX	;
7409	DD23	INC	IX	;
740B	DD	DEFB	000H	; LSB von IX in Akku
740C	7D	LD	A, L	;
740D	FE68	CP	68H	; = 68H ? (IX = 7858H)
740F	C2D373	JP	NZ, 73D3H	; wenn nicht, nächste Explosion
7412	2A0242	LD	HL, (4202H)	; Punktzahl
7415	11DE79	LD	DE, 79DEH	; auf 79DEH (Ausgabezeile)
7418	CD2C73	CALL	732CH	; berechnen und ausgeben
741B	2A0242	LD	HL, (4202H)	; Punktzahl
741E	ED5B0442	LD	DE, (4204H)	; und bestes Ergebnis
7422	DF	RST	18H	; vergleichen
7423	3809	JR	C, \$+0BH	; wenn Punktzahl kleiner, dann nach 742EH
7425	220442	LD	(4204H), HL	; sonst als bestes Ergebnis abspeichern
7428	11FB79	LD	DE, 79FBH	; und auf 79FBH (Ausgabezeile)
742B	CD4F73	CALL	734FH	; berechnen und ausgeben
742E	C9	RET		; zuruck

*** UNTERPROGRAMM SCHIFF TESTEN UND LÖSCHEN ***

7438	2A0642	LD	HL, (4206H)	; Schiffposition in HL
743B	0603	LD	B, 03H	; Zähler fuer Anzahl der Bytes
743D	7E	LD	A, (HL)	; Zeichen aus Bildeichn in Akku
743E	FEA0	CP	0A0H	; (A0H ? (gehört zu Feinden)
7440	3809	JR	C, \$+0BH	; wenn ja, dann nach 744BH
7442	3620	LD	(HL), 20H	; sonst löschen

7444	23	INC	HL	; naechstes Byte
7445	10F6	DJNZ	\$-08H	; alle getestet ? wenn nicht, nach 743DH
7447	2B	DEC	HL	; HL wieder auf die Position des
7448	2B	DEC	HL	; ersten Zeichens bringen
7449	2B	DEC	HL	;
744A	C9	RET		; und zurueck
744B	F1	POP	AF	; Stack korrigieren
744C	C33075	JP	7530H	; Unterprg.: Explosion des Schiffes

; *** SCHIFF UND FARREN SETZEN ***

7450	06A0	LD	B, 0A0H	; A0H ist erste Phase des Schiffes
7452	3A0A42	LD	A, (420AH)	; Zaehler in Akku
7455	E601	AND	01H	; undieren mit 01H (ergibt 0 oder 1)
7457	2802	JR	Z, \$+04H	; = 0 ? dann nach 745BH
7459	06A3	LD	B, 0A3H	; sonst A3H in B (zweite Phase)
745B	78	LD	A, B	; B in Akku
745C	77	LD	(HL), A	; auf Schiffposition
745D	23	INC	HL	; naechste Adresse
745E	3C	INC	A	; naechstes Zeichen
745F	77	LD	(HL), A	; abspeichern
7460	23	INC	HL	; naechste Adresse
7461	3C	INC	A	; naechstes Zeichen
7462	77	LD	(HL), A	; abspeichern
7463	11FEAB	LD	DE, 0ABFEH	; Offset zum Farbspeicher -2
7466	19	ADD	HL, DE	; zur Position addieren
7467	3606	LD	(HL), 06H	; 06H abspeichern (orange)
7469	23	INC	HL	; naechste Adresse
746A	3602	LD	(HL), 02H	; 02H abspeichern (rot)
746C	23	INC	HL	; naechste Adresse
746D	3606	LD	(HL), 06H	; 06H abspeichern (orange)
746F	C9	RET		; und Zurueck

; *** BILDAUFBAU UND SPIELSTART ***

7480	210044	LD	HL, 4400H	; Erste Bildschirmposition
7483	110144	LD	DE, 4401H	; Zweite Bildschirmposition
7486	01FF03	LD	BC, 03FFH	; Bildschirmmaenge -1
7489	3620	LD	(HL), 20H	; 20H auf erstes Zeichen (SPACE)
748B	EDB0	LDIR		; 1. Zeichen auf 2. schieben usw.
748D	21D879	LD	HL, 79D8H	; Ausgabezeile
7490	110044	LD	DE, 4400H	; auf erste Bildschirmzeile
7493	012800	LD	BC, 0028H	; 400 Zeichen
7496	EDB0	LDIR		; schieben
7498	210078	LD	HL, 7800H	; Farben der ersten Zeile
749B	1100F0	LD	DE, 0F000H	; in den Farbspeicher
749E	012800	LD	BC, 0028H	; 400 Bytes
74A1	EDB0	LDIR		; schieben
74A3	3A0142	LD	A, (4201H)	; Feindtyp in Akku
74A6	3C	INC	A	; +1

74A7	E607	AND	07H	; mit 07H undieren (nur 0..7 zugel.)
74A9	320142	LD	(4201H), A	; und wieder abspeichern
74AC	21AB47	LD	HL, 47ABH	; Schiffsposition, vorletzte Zeile, Mitte
74AF	220642	LD	(4206H), HL	; abspeichern
74B2	210079	LD	HL, 7300H	; Tabelle der Feindpositionen
74B5	118078	LD	DE, 7880H	; in die Arbeitstabelle
74B8	018000	LD	BC, 0080H	; 128D Bytes (fuer 32D Feinde)
74BB	EDB0	LDIR		; schieben
74BD	210100	LD	HL, 0001H	; X-Richtung (0001H-rechts, FFFFH-links)
74C0	220842	LD	(4208H), HL	; abspeichern
74C3	215078	LD	HL, 7850H	; Schuesse und Explosionen
74C6	115178	LD	DE, 7851H	; mit 00H ueberschreiben
74C9	012F00	LD	BC, 002FH	; (0000H als Position bedeutet Schuss,
74CC	3600	LD	(HL), 00H	; Explosion oder Feind existiert nicht)
74CE	EDB0	LDIR		;
74D0	3E20	LD	A, 20H	; Anzahl der Feinde (32D) abspeichern
74D2	320B42	LD	(420BH), A	;
74D5	3A0042	LD	A, (4200H)	; Level in Akku
74D8	3C	INC	A	; +1
74D9	320042	LD	(4200H), A	; und wieder abspeichern
74DC	212878	LD	HL, 7828H	; Text '*** GET READY **'
74DF	113C46	LD	DE, 463CH	; in Bildschirmspeicher
74E2	011000	LD	BC, 0010H	; 16 Zeichen
74E5	EDB0	LDIR		; schieben
74E7	2A0642	LD	HL, (4206H)	; Schiffsposition in HL
74EA	CD5074	CALL	7450H	; Schiff und Farben setzen
74ED	3E18	LD	A, 18H	; 18H = JR unbedingt, damit die Feinde
74EF	328971	LD	(7189H), A	; im Vorspann keine Y-Richtung bekommen
74F2	060C	LD	B, 0CH	; Zaehler (fuer 12D Durchlaeufer)
74F4	C5	PUSH	BC	; retten
74F5	CD0071	CALL	7100H	; Unterprg.: Feinde bewegen
74F8	CD071	CALL	7100H	; Unterprg.: X-Richtung testen
74FB	213CF2	LD	HL, 0F23CH	; Farbspeicheradresse von 'GET READY'
74FE	113DF2	LD	DE, 0F23DH	; mit einem
7501	010F00	LD	BC, 000FH	; zufaelligen Wert
7504	ED5F	LD	A, R	; aus dem R-(Refresh-) Register
7506	77	LD	(HL), A	; belegen
7507	EDB0	LDIR		; 16D Bytes
7509	010018	LD	BC, 1800H	; Warteschleifenwert
750C	CD6000	CALL	0060H	; ROM-Routine: Schleife auf BC
750F	C1	POP	BC	; BC zurueck vom Stack
7510	10E2	DJNZ	\$-1CH	; und evtl. neuer Durchlauf
7512	3E20	LD	A, 20H	; 20H = JR NZ (non zero), die Feinde
7514	328971	LD	(7189H), A	; koennen sich jetzt wieder voll bewegen
7517	213C46	LD	HL, 463CH	; '*** GET READY **' Position im Bild-
751A	113D46	LD	DE, 463DH	; schirmspeicher mit 20H (SPACE)
751D	010F00	LD	BC, 000FH	; belegen
7520	77	LD	(HL), A	; im Akku steht noch 20H
7521	EDB0	LDIR		; 16D Bytes
7523	C30072	JP	7200H	; Sprung zur Hauptschleife

; *** EXPLOSION DES SCHIFFES ***

7530	061F	LD	B, 1FH	; Zaehler (31 Durchlaeufer)
------	------	----	--------	-----------------------------

7532	3E07	LD	A,07H	; 07H (SOUND 7,23)
7534	D3F8	OUT	(0F8H),A	; auf Port FBH Kanal 1 und 2 werden
7536	3EE7	LD	A,0E7H	; E7H auf Rauschgenerator
7538	D3F9	OUT	(0F9H),A	; auf Port F9H geschaltet)
753A	C5	PUSH	BC	; Zaehler retten
753B	3E09	LD	A,09H	; 09H (SOUND 9,0..15
753D	D3F8	OUT	(0F8H),A	; auf Port FBH Lautstaerke von
753F	78	LD	A,B	; Zaehler in Akku Kanal 2 wird mit
7540	E60F	AND	0FH	; mit 0FH undieren den Schleifenzaehler
7542	D3F9	OUT	(0F9H),A	; auf Port F9H gesendert)
7544	3E06	LD	A,06H	; 06H
7546	D3F8	OUT	(0F8H),A	; auf Port FBH (SOUND 6,0..3)
7548	3E1F	LD	A,1FH	; 1FH in Akku Rauschfrequenz wird
754A	90	SUB	B	; minus Zaehler mit dem Zaehler ge-
754B	D3F9	OUT	(0F9H),A	; auf Port F9H sendert)
754D	3A0A42	LD	A,(420AH)	; Zaehler in Akku
7550	3C	INC	A	; +1
7551	320A42	LD	(420AH),A	; und wieder abspeichern
7554	CD0071	CALL	7100H	; Unterprg.: Peinde bewegen
7557	CD071	CALL	71C0H	; Unterprg.: X-Richtung testen
755A	CD8073	CALL	7380H	; Unterprg.: Schuesse und Explosionen
755D	2A5078	LD	HL,(7850H)	; Position der ersten Explosion in HL
7560	7C	LD	A,H	; MSB
7561	B5	OR	L	; oderiert mit LSB
7562	2018	JR	NZ,\$+1AH	; (0 ? dann nach 757CH
7564	2A0642	LD	HL,(4206H)	; Schiffposition in HL
7567	ED5F	LD	A,R	; R-Register als Zufallswert
7569	E601	AND	01H	; mit 01H undieren
756B	2801	JR	Z,\$+03H	; = 0 ? dann nach 756EH
756D	23	INC	HL	; sonst Position +1 (rechter Teil)
756E	225078	LD	(7850H),HL	; in Explosionstabelle abspeichern
7571	325278	LD	(7852H),A	; Akku in Explosionszaehler
7574	3E08	LD	A,08H	; 08H (SOUND 8,15
7576	D3F8	OUT	(0F8H),A	; auf Port FBH Kanal 1 auf maximale
7578	3E0F	LD	A,0FH	; 0FH Lautstaerke, Explo-
757A	D3F9	OUT	(0F9H),A	; auf Port F9H sionegeraueuch)
757C	C1	POP	BC	; Schleifenzaehler zurueck
757D	10BB	DJNZ	\$-43H	; fertig ? wenn nicht, dann nach 753AH
757F	21EE79	LD	HL,79EEH	; Letztes Schiff in der Ausgabeseile
7582	7E	LD	A,(HL)	; Zeichen in Akku
7583	FEB0	CP	0B0H	; = B0H (Zeichen fuer Schiff)
7585	2B09	JR	Z,\$+0BH	; wenn ja, dann nach 7590H
7587	2B	DEC	HL	; ein Zeichen vor
7588	7D	LD	A,L	; LSB in Akku
7589	FEE3	CP	0E3H	; = Position des ersten Schiffes -1 ?
758B	20F5	JR	NZ,\$-09H	; wenn nicht, dann nach 7582H
758D	C3A075	JP	75A0H	; sonst Spielende
7590	3620	LD	(HL),20H	; Schiff in Ausgabeseile loeschen
7592	3A0142	LD	A,(4201H)	; Peindtyp in Akku
7595	3D	DEC	A	; -1
7596	320142	LD	(4201H),A	; wieder abspeichern
7599	C38074	JP	7480H	; und das Bild wieder aufbauen

; *** SPILENDE ***

75A0	21FA79	LD	HL,79FAH	; bestes Ergebnis von der Ausgabeweile
75A3	11607D	LD	DE,7D60H	, in die Anfangsgrafik kopieren
75A6	010600	LD	BC,0006H	,
75A9	ED80	LDIR		,
75AB	C30070	JP	7000H	, Zur Eingabeschleife und neues Spiel

; *** PROGRAMMSTART ***

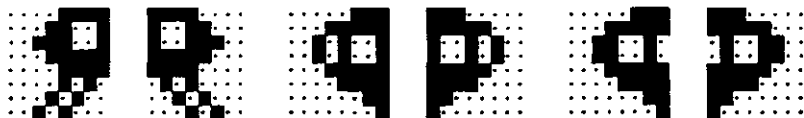
75B0	210000	LD	HL,0000H	, High-Score
75B3	220442	LD	(4204H),HL	, löschen
75B6	310060	LD	SP,6000H	, Stackpointer auf 6000H (vor Programm)
75B9	C30070	JP	7000H	, und Sprung zur Eingabeschleife

**** DER ZEICHENSATZ DES PROGRAMMS ZALAGA ****

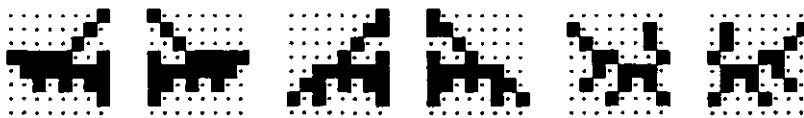
CHR\$(128) CHR\$(129) CHR\$(130) CHR\$(131) CHR\$(132) CHR\$(133)



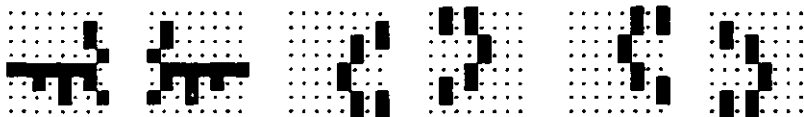
CHR\$(134) CHR\$(135) CHR\$(136) CHR\$(137) CHR\$(138) CHR\$(139)



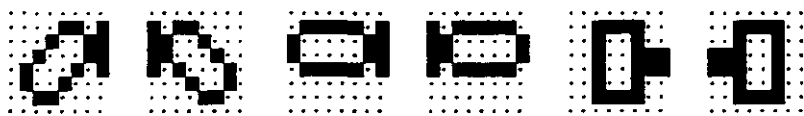
CHR\$(140) CHR\$(141) CHR\$(142) CHR\$(143) CHR\$(144) CHR\$(145)



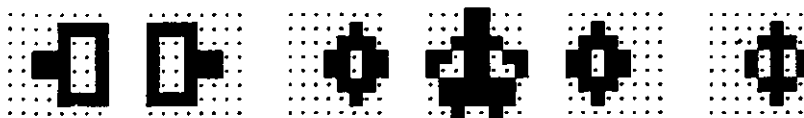
CHR\$(146) CHR\$(147) CHR\$(148) CHR\$(149) CHR\$(150) CHR\$(151)



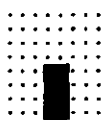
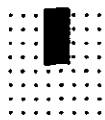
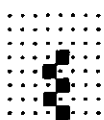
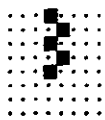
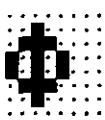
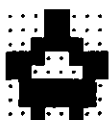
CHR\$(152) CHR\$(153) CHR\$(154) CHR\$(155) CHR\$(156) CHR\$(157)



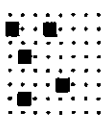
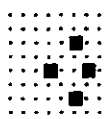
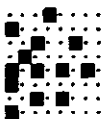
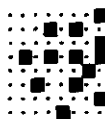
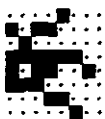
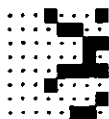
CHR\$(158) CHR\$(159) CHR\$(160) CHR\$(161) CHR\$(162) CHR\$(163)



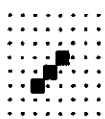
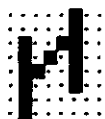
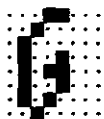
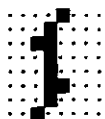
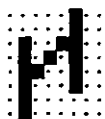
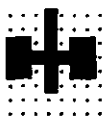
CHR\$(164) CHR\$(165) CHR\$(166) CHR\$(167) CHR\$(168) CHR\$(169)



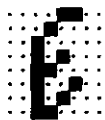
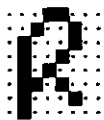
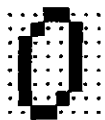
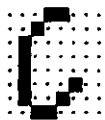
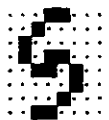
CHR\$(170) CHR\$(171) CHR\$(172) CHR\$(173) CHR\$(174) CHR\$(175)



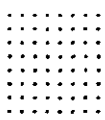
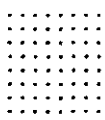
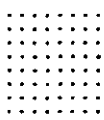
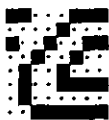
CHR\$(176) CHR\$(177) CHR\$(178) CHR\$(179) CHR\$(180) CHR\$(181)



CHR\$(182) CHR\$(183) CHR\$(184) CHR\$(185) CHR\$(186) CHR\$(187)



CHR\$(188) CHR\$(189) CHR\$(190) CHR\$(191) CHR\$(192) CHR\$(193)



Ein schnelles Formatierprogramm für das Colour-Genie
(nur für Diskettenbenutzer)

Sicher ist Ihnen nicht entgangen, daß die Formatieroutine des Colour-DOS einen sehr großen Nachteil hat: sie ist zu langsam! Wir präsentieren hier ein Programm, das Disketten auf dem Colour-Genie in einem Bruchteil der gewohnten Zeit formatiert. Dieses Programm besteht aus zwei Teilen:

1. Ein Maschinenspracheprogramm. Geben Sie dieses bitte auf irgendeine Art (z.B. mit dem Basic-Monitor, den Sie weiter vorne in diesem Buch gelistet finden; für diesen sind auch die Prüfsummen am Ende jeder Zeile bestimmt, diese also nicht (!) mit eingeben.) in den Rechner ein. Dann speichern Sie dieses Programm mit
CMD"\\ FFORMAT/CMD,&H8000,&H82A7,&H0066"
auf Diskette ab. Geben Sie nun das kleine BASIC-Programm ein, das hinter dem Hex-Listing steht. Dieses Programm lädt und startet dann das Formatier-Programm.

```
8000 2B 3E 2C 20 3C 3B 3E 2C 96
8008 20 3C 40 3E 2C 20 3C 53 B5
8010 70 61 28 4E 16 4E 0C 00 B7
8018 03 F5 00 01 16 4E 0C 00 69
8020 03 F5 00 00 14 FF 0B FF 15
8028 06 00 00 01 0B FF 06 00 17
8030 00 00 05 C5 3A 0E 5A CB 07
8038 3F 47 57 21 00 80 7A 90 88
8040 77 23 82 77 23 10 F7 21 DE
8048 00 80 09 11 A8 B2 21 14 C9
8050 80 3A 11 5A E6 01 20 03 2F
8058 21 26 80 2B 7E 32 AE 80 D0
8060 2B CD B8 80 3A 0E 5A 4F 21
8068 E5 E1 E5 CD B8 80 20 FB CB
8070 EB 36 FE 23 3A A7 82 77 1C
8078 23 36 00 23 D9 7E 23 D9 CF
8080 77 23 36 01 23 36 F7 23 44
8088 EB CD B8 80 20 FB EB 36 2C
8090 FB 23 06 00 36 E5 23 10 72
8098 FB 36 F7 23 EB 0D C2 69 6E
80A0 80 21 A8 A2 AF ED 52 23 FC
80A8 44 4D D5 E1 13 36 4E ED CB
80B0 B0 E1 21 A8 B2 C1 D1 C9 37
80B8 C5 46 23 4E 23 78 B7 28 F6
80C0 07 79 12 13 10 FC AF 3D 9D
80C8 C1 C9 E5 D5 C5 CD BB D0 61
80D0 11 EF FF 21 EC FF 01 A8 B4
80D8 82 36 F0 CD 37 D1 0A CB 52
80E0 46 2B 27 CB 4E 28 F8 12 E0
```

80E8	03	0A	CB	4E	CA	EA	80	12	6C
80F0	03	0A	CB	4E	20	F9	CB	4E	58
80F8	20	F5	CB	4E	20	F1	CB	46	50
8100	28	08	CB	4E	20	E9	CB	7E	98
8108	28	E8	7E	36	D0	E6	F8	C1	33
8110	D1	E1	C9	3A	A7	82	21	E3	E2
8118	81	CD	05	82	21	D3	81	ED	37
8120	5B	20	40	01	0A	00	ED	B0	63
8128	21	DD	81	01	13	00	ED	B0	30
8130	CD	32	80	06	03	CD	CA	80	9F
8138	28	14	10	F9	47	3E	10	CB	A5
8140	78	C2	14	82	3D	CB	70	C2	0A
8148	14	82	3D	C3	14	82	21	F0	3D
8150	81	ED	5B	20	40	01	0A	00	34
8158	ED	B0	3A	0E	5A	47	11	00	97
8160	80	C5	1A	21	03	82	CD	05	D7
8168	82	2A	20	40	D5	11	12	00	04
8170	19	EB	21	FA	81	01	0B	00	AC
8178	ED	B0	D1	06	03	21	A8	82	C2
8180	1A	CD	5B	82	E6	1C	28	06	F4
8188	10	F3	C1	C3	13	81	13	C1	EF
8190	10	CF	C9	CD	99	00	AF	32	BF
8198	A7	82	32	ED	FF	3A	0D	5A	EB
81A0	47	C5	3A	0A	5A	E6	40	28	F8
81A8	08	21	09	5A	CB	E6	CD	13	1D
81B0	81	21	09	5A	CB	A6	CD	13	56
81B8	81	0E	43	CD	9B	D0	3A	A7	EB
81C0	82	3C	32	A7	82	32	ED	FF	37
81C8	C1	10	D6	3E	0D	CD	33	00	F2
81D0	C3	99	D0	46	6F	72	6D	61	21
81D8	74	69	6E	67	20	54	72	61	F9
81E0	63	68	20	30	30	20	20	20	AE
81E8	20	20	20	20	20	20	20	20	00
81F0	56	65	72	69	66	79	69	6E	4C
81F8	67	20	2C	20	53	65	63	74	62
8200	6F	72	20	30	30	C5	06	2F	5B
8208	04	D6	0A	30	FB	C6	3A	70	7F
8210	23	77	C1	C9	6F	26	00	CD	86
8218	9A	0A	ED	7B	20	82	E1	C9	58
8220	00	00	E5	ED	73	20	82	CD	B4
8228	7F	0A	7D	CD	40	D1	C2	14	BA
8230	82	F3	3E	C9	32	12	40	AF	AF
8238	32	23	40	ED	7B	81	40	CD	88
8240	C9	01	CD	93	81	AF	32	ED	79
8248	FF	11	00	00	CD	03	DE	21	DF
8250	00	00	CD	9A	0A	ED	7B	20	F9
8258	82	E1	C9	E5	D5	C5	E5	C1	51
8260	F5	11	EF	FF	21	EC	FF	CD	CD
8268	BB	D0	F1	32	EE	FF	3A	A7	7C
8270	82	32	ED	FF	36	89	CD	37	62
8278	D1	CB	46	28	21	3E	83	A6	92

```

8280 E2 7D 82 1A 02 03 CB 4E 19
8288 20 F9 CB 4E 20 F5 CB 4E 60
8290 20 F1 CB 46 28 08 CB 4E 6B
8298 20 E9 CB 7E 28 E8 7E 36 16
82A0 D0 E6 3C C1 D1 E1 C9 00 2E
82A8

```

Und hier das dazugehoerige BASIC-Programm:

```

10 CLS:PRINT"Fast Format":PRINT:PRINT
20 INPUT"Welches Laufwerk";A
30 CMD"L FFORMAT/CMD"
40 DEFUSR1=&H0222
50 X=USR1(A)
60 IFX<>0THENPRINT"Disk error No.":X
70 END

```

Erläuterung des Maschinenprogramms "FAST FORMAT"

Das Programm wird mit dem `USR`-Befehl aufgerufen. Es wird zur Routine `MAIN` gesprungen. Dort wird das `HL`-Register (der Zeiger auf den Programtext) gerettet. Die Routine `0A7FH` liest das Argument des `USR`-Befehls als Integerzahl in das `HL`-Register, interessant ist davon aber nur das niederwertige Byte, die Laufwerksnummer, in der die zu formatierende Diskette liegt. Die Routine `MOUNT` (eine `DOS`-Routine, die bei `D140H` beginnt), selektiert dieses Laufwerk und überprüft, ob eine Diskette in diesem Laufwerk liegt und die Laufwerksklappe geschlossen ist. Sollte dies nicht der Fall sein, wird zur Fehlerbehandlungsroutine gesprungen. Sonst werden die Interrupts abgeschaltet, der Bildschirm gelöscht (`01C9H`) und die eigentliche Formatroutine aufgerufen. Zum Schluß richtet das `DOS` das Inhaltsverzeichnis der Diskette ein (`CALL 0DE03H`). Wichtig: Beim Sprung auf diese Routine müssen das `A`, `E` und das `D`-Register den Wert Null enthalten. Schließlich, wenn alles gutgegangen ist, wird die Fehlernummer Null (kein Fehler) durch `CALL 0A9AH` zurückgegeben. Nun die Erläuterung der einzelnen Unterprogramme:

BUILD: Dieses Programm baut ab der Adresse `BUFFER` den zu schreibenden Track auf. Das Aussehen des Tracks ist variabel, es hängt von der Einstellung für dieses Laufwerk ab:

SINGLE DENSITY:

	20 Bytes	0FFH
pro Sektor:	11 Bytes	0FFH
	6 Bytes	00H
	1 Byte	0FEH
	1 Byte	Spurnummer
	1 Byte	Seitennummer (immer Null)
	1 Byte	Sektornummer
	1 Byte	Sektorlänge (siehe unten)
	1 Byte	0F7H (generiert Prüfsumme)
	11 Bytes	0FFH
	6 Bytes	00H
	1 Byte	0FBH
	n Bytes	0E5H (n ist Sektorlänge)
	1 Byte	0F7H (generiert Prüfsumme)

Diese Daten werden für jeden Sektor geschrieben.
Dann folgen Bytes `0FFH`, bis die gesamte Spur `2000H` Bytes lang ist.

DOUBLE DENSITY:

	40 Bytes 4EH
pro Sektor:	22 Bytes 4EH
	12 Bytes 00H
	3 Bytes 0F5H
	1 Byte 0FEH
	1 Byte Spurnummer
	1 Byte Seitennummer (inner Null)
	1 Byte Sektornummer
	1 Byte Sektorlänge (siehe unten)
	1 Byte 0F7H (generiert Prüfsumme)
	22 Bytes 4EH
	12 Bytes 00H
	3 Bytes 0F5H
	1 Byte 0FBH
	n Bytes Daten (n = Sektorlänge)
	1 Byte 0F7H (generiert Prüfsumme)

Diese Daten werden für jeden Sektor geschrieben, dann folgen Bytes 4EH, bis die Spur 2000H Bytes im Speicher belegt.

Die Sektorlänge bedeutet folgendes:

00	= 128 Bytes
01	= 256 Bytes
02	= 512 Bytes
03	= 1024 Bytes

WATTRK: Die Routine WATTRK schreibt den vorher vorbereiteten Speicherbereich ab BUFFER auf die Spur, auf der momentan der Schreib-/Lesekopf des Laufwerks steht.

FMSUB: Dieses Programm baut eine Spur im Speicher auf, schreibt sie auf die Diskette und versucht dann, jeden einzelnen Sektor wieder einzulesen, um sicherzustellen, daß die Formatierung einwandfrei verlaufen ist.

FORMAT: FORMAT formatiert eine ganze Diskette. Die Routine FMSUB wird für jede einzelne Spur aufgerufen, dann wird der Schreib-/Lesekopf des Laufwerks eine Spur weiter nach innen gesetzt. Dies wird sooft wiederholt, wie die Diskette Spuren hat.

RDSEC: RDSEC liest einen Sektor von der Spur, auf der der Schreib-/Lesekopf momentan steht.


```

org #000h
sectab: defs 18
        defb 40,4eh
secdd:  defb 22,4eh,12,0,3,0f5h,0,1,22,4eh,12,0,3,0f5h,0,0
        defb 20,0ffh
secsd:  defb 11,0ffh,6,0,0,1,11,0ffh,6,0,0,0
build:  push de
        push bc
        ld a,(5a0eh)
        srl a
        ld b,a
        ld d,a
        ld hl,sectab
by:      ld a,d
        sub b
        ld (hl),a
        inc hl
        add a,d
        ld (hl),a
        inc hl
        djnz by
        ld hl,sectab
        exx
        ld de,buffer
        ld hl,secdd
        ld a,(5a11h)
        and 1
        jr nz,bx
        ld hl,secsd
bx:      dec hl
        ld a,(hl)
        ld (filler),a
        dec hl
        call wrtstr
        ld a,(5a0eh)
        ld c,a
        push hl
build1:  pop hl
        push hl
build2:  call wrtstr
        jr nz,build2
        ex de,hl
        ld (hl),0feh
        inc hl
        ld a,(trksec)
        ld (hl),a
        inc hl
        ld (hl),0
        inc hl
        exx
        ld a,(hl)
        inc hl
        exx
        ld (hl),a
        inc hl

```

```

        ld    (hl),1
        inc  hl
        ld    (hl),0f7h
        inc  hl
        ex    de,hl
build3: call wrtstr
        jr    nz,build3
        ex    de,hl
        ld    (hl),0fbh
        inc  hl
        ld    b,0
build4: ld    (hl),0e5h
        inc  hl
        djnz build4
        ld    (hl),0f7h
        inc  hl
        ex    de,hl
        dec  c
        jp    nz,build4
        ld    hl,buffer+2000h
        xor  a
        sbc  hl,de
        inc  hl
        ld    b,h
        ld    c,l
        push de
        pop  hl
        inc  de
        ld    (hl),04eh
filler  equ  $-1
        ldir
        pop  hl
        ld    hl,buffer
        pop  bc
        pop  de
        ret
wrtstr: push  bc
        ld    b,(hl)
        inc  hl
        ld    c,(hl)
        inc  hl
        ld    a,b
        or   a
        jr    z,wrtst2
        ld    a,c
wrtst1: ld    (de),a
        inc  de
        djnz wrtst1
        xor  a
        dec  a
wrtst2: pop  bc
        ret
wrttrk: push  hl
        push de
        push bc

```

```

call reset+5
ld de,0ffefh
ld hl,0ffech
ld bc,buffer
ld (hl),0f0h
call wait
ld a,(bc)
wrt1: bit 0,(hl)
      jr z,wrt3
      bit 1,(hl)
      jr z,wrt1
      ld (de),a
      inc bc
      ld a,(bc)
wrt11: bit 1,(hl)
      jp z,wrt11
wrt12: ld (de),a
      inc bc
      ld a,(bc)
wrt13: bit 1,(hl)
      jr nz,wrt12
      bit 1,(hl)
      jr nz,wrt12
      bit 1,(hl)
      jr nz,wrt12
      bit 0,(hl)
      jr z,wrt3
      bit 1,(hl)
      jr nz,wrt12
      bit 7,(hl)
      jr z,wrt13
wrt3:  ld a,(hl)
      ld (hl),0d0h
      and 0f8h
      pop bc
      pop de
      pop hl
      ret
fmsub: ld a,(trksec)
      ld hl,str3
      call convrt
      ld hl,str1
      ld de,(4020h)
      ld bc,10
      ldir
      ld hl,str2
      ld bc,19
      ldir
      call build
      ld b,3
form2: call wrttrk
      jr z,form3
      djnz form2
      ld b,a
      ld a,16

```

```

        bit    7,b
        jp     nz,dskerr
        dec    a
        bit    6,b
        jp     nz,dskerr
        dec    a
        jp     dskerr
form3:  ld     hl,str4
        ld     de,(4020h)
        ld     bc,10
        ldir
        ld     a,(5a0eh)
        ld     b,a
        ld     de,sectab
form4:  push    bc
        ld     a,(de)
        ld     hl,str5+9
        call   convrt
        ld     hl,(4020h)
        push   de
        ld     de,18
        add    hl,de
        ex     de,hl
        ld     hl,str5
        ld     bc,11
        ldir
        pop    de
        ld     b,3
form5:  ld     hl,buffer
        ld     a,(de)
        call   rdsec
        and    1ch
        jr     z,form6
        djnz   form5
        pop    bc
        jp     fmsub
form6:  inc     de
        pop    bc
        djnz   form4
        ret
format: call   restor
        xor    a
        ld     (trksec),a
        ld     (0ffedh),a
        ld     a,(5a0dh)
        ld     b,a
form1:  push    bc
        ld     a,(5a0ah)
        and    40h
        jr     z,form11
        ld     hl,5a09h
        set    4,(hl)
        call   fmsub
form11: ld     hl,5a09h
        res    4,(hl)

```

```

        call fmsub
        ld c,43h
        call isscmd
        ld a,(trksec)
        inc a
        ld (trksec),a
        ld (Offedh),a
        pop bc
        djnz form1
        ld a,13
        call 33h
        jp restor
str1:   defm 'Formating '
str2:   defm 'Track '
str3:   defm '00
str4:   defm 'Verifying '
str5:   defm ', Sector 00'
convrt: push bc
        ld b,2fh
conv1:  inc b
        sub 10
        jr nc,conv1
        add a,3ah
        ld (hl),b
        inc hl
        ld (hl),a
        pop bc
        ret
diskerr: ld l,a
        ld h,0
        call 0a9ah
        ld sp,(stack)
        pop hl
        ret
stack:  defw 0
main:   push hl
        ld (stack),sp
        call 0a7fh
        ld a,l
        call mount
        jp nz,dskerr
        di
        ld a,0c9h
        ld (4012h),a
        xor a
        ld (4023h),a
        ld sp,(40b1h)
        call 01c9h
        call format
        xor a
        ld (Offedh),a
        ld de,0
        call 0de03h
        ld hl,0
        call 0a9ah

```

```

        ld    sp,(stack)
        pop   hl
        ret
rdsec:  push  hl
        push  de
        push  bc
        push  hl
        pop   bc
        push  af
        ld    de,0ffefh
        ld    hl,0ffech
        call  resel+5
        pop   af
        ld    (0ffeeh),a
        ld    a,(trksec)
        ld    (0ffedh),a
        ld    (hl),88h
        call  wait
        bit   0,(hl)
        jr    z,rdsc3
rdsc0:  ld    a,83h
        and   (hl)
        jp    po,rdsc0
rdsc1:  ld    a,(de)
        ld    (bc),a
        inc   bc
rdsc2:  bit   1,(hl)
        jr    nz,rdsc1
        bit   1,(hl)
        jr    nz,rdsc1
        bit   1,(hl)
        jr    nz,rdsc1
        bit   0,(hl)
        jr    z,rdsc3
        bit   1,(hl)
        jr    nz,rdsc1
        bit   7,(hl)
        jr    z,rdsc2
rdsc3:  ld    a,(hl)
        ld    (hl),0d0h
        and   3ch
        pop   bc
        pop   de
        pop   hl
        ret
trksec: defb 0
select equ 0d0c7h
resel   equ 0d0b6h
wait    equ 0d137h
isscmd  equ 0d09bh
restor  equ 0d099h
mount   equ 0d140h
buffer  equ $
end     main

```

Sektoren lesen und schreiben - vom Basic aus

Dieses kleine Programm können Sie in Ihre eigenen BASIC-Programme einbauen. Lassen Sie dann die Zeilen 30 bis 60 weg und übergaben Sie die in der REM-Zeile genannten Argumente an das Programm. Zum Programm noch folgendes:

Das eigentliche Programm ist ein kurzes Maschinenspracheprogramm, das in A\$ eingelesen wird. Es sieht folgendermaßen aus:

```
LD    HL,adresse
LD    DE,disk-relativer-sektor
CALL  unterprogramm
LD    L,A
LD    H,0
JP    0A9AH
```

Die Werte für adresse, disk-relativer-sektor und unterprogramm werden dann entsprechend Ihren Eingaben gePOKEd. Dieses Programm liest bzw. schreibt dann einen Sektor und gibt als Ergebnis des USR-Aufrufs die Nummer des aufgetretenen Fehlers an (0 = kein Fehler).

Programmlisting:

```
10 REM **** AD=address, SN=sector #, FL: 1=read, 2=write, 3=wp, DR=Laufwerksnummer
20 CLS:PRINT"Sektor In-/Output":PRINT:PRINT
30 INPUT"Laufwerk (0-3)":DV:IFDV<0ORDV>3THEN30
40 INPUT"Adresse":AD
50 INPUT"Diskrelativer Sektor":SN
60 INPUT"(1) Lesen, (2) Schreiben, (3) Lese-      geschuetzt schreiben"
:FL
70 RESTORE
80 READA$:IFA$<>"SECIO"THEN80
90 A$="":FORX=1TO15:READA:A$=A$+CHR$(A):NEXTX:FORX=1TOFL:READDR:NEXTX
100 X=VARPTR(A$):X=PEEK(X+1)+256*PEEK(X+2):IFX>32767THENX=X-65536
110 POKEX+1,(AD AND 255):POKE X+2,(INT(AD/256)AND255)
120 POKEX+4,(SN AND 255):POKE X+5,(INT(SN/256)AND255)
130 POKEX+7,(DR AND 255):POKE X+8,(INT(DR/256)AND255)
140 POKE&H5A08,DV:DEFUSR1=X:X=USR1(0):IFX=0THENEND
150 PRINT"Disk Error No.":X:END
160 DATASECIO
170 DATA33,0,0,17,0,0,205,0,0,111,38,0,195,154,10
180 DATA-12433,-12417,-12421
```

Zwei Tips für den Betrieb eines einfarbigen Monitors als Colour-Genie

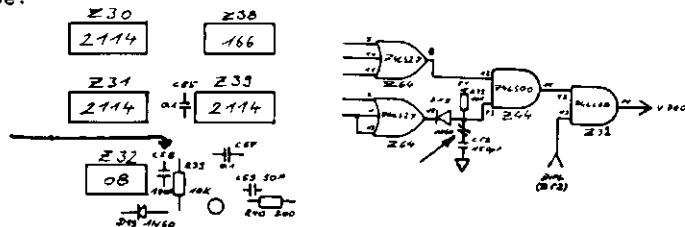
Viele Colour-Genie-Benutzer haben an Ihren Computer einen einfarbigen Monitor angeschlossen, da diese billiger als ein RGB-Farbmonitor sind, aber ein besseres Bild liefern, als ein Fernseher.

In diesem Falle kann man die Schrift noch verbessern, indem man

- 1.) Alle Farben auf hellweiß umgesetzt (COLOUR 16).
Dies geht bei Basic-Programmen sehr einfach:
In den Adressen 4390H bis 439FH legt das Basic eine Farb-Tabelle an, die festlegt, welche Farbe durch welchen Colour-Befehl gewählt wird. Alles was Sie nun machen müssen, ist in diese Tabelle überall den 15 POKEn (Dies entspricht Hellweiß, s. auch Anhang C).
Also:
FORA=&H4390 TO &H439F : POKE A, 15 : NEXT
Bei Maschinenspracheprogrammen und Programmen, die direkt in den Farbspeicher POKEn, hilft dies i.d.R. nicht.
- 2.) Eine kleinen Geräteumbau vornimmt:
In der Video-Schaltung des Colour-Genies werden die Matrix-Punkte aller Zeichen verbreitert, damit bei Betrieb mit einem Farb-Fernseher die Punkte groß genug sind, um von der Lochmaskenrohre aufgelöst werden zu können.
Dies ist bei einem Monitor natürlich überflüssig.
Diese Punktverbreiterung kann durch Entfernen eines Kondensators rückgängig gemacht werden.
Dazu gehen Sie folgendermaßen vor:

Lösen Sie die 3 Schrauben vorne unten und öffnen Sie das Colour-Genie - Achten Sie darauf, daß Sie das Kabel des Level-Meters (wenn eingebaut) und das der Tastatur nicht abziehen.

Sie sehen vorne, in der Mitte der Hauptplatine, diese Bauteilgruppe:



Schneiden Sie nun mit einem feinen Seitenschneider eines der Anschlußbeine des Kondensators C58 durch und biegen Sie den Kondensator etwas hoch. (So können Sie den Kondensator jederzeit wieder anloten.)

Schrauben Sie das Gerät wieder zu. Fertig!

(Etwas Elektronik-Erfahrung sollten Sie für diesen Umbau haben - sonst lassen Sie einen kundigen Bekannten die Umrüstung vornehmen.)

Anhang A Colour-Basic-Tokens

Basic Einfachtokens

128	↵	END	129	┐	FOR
130	┐	RESET	131	┐	SET
132	┐	CLS	133	┐	CMD
134	▲	RANDOM	135	┐	NEXT
136	┐	DATA	137	┐	INPUT
138	┐	DIM	139	┐	READ
140	┐	LET	141	□	GOTO
142	┐	RUN	143	·	IF
144	■	RESTORE	145	■	GOSUB
146	┐	RETURN	147	┐	REM
148	➤	STOP	149	◀	ELSE
150	┐	TRON	151	┐	TROFF
152	■	DEFSTR	153	┐	DEFINT
154	┐	DEFSNG	155	┐	DEFDBL
156	┐	LINE	157	▲	EDIT
158	┐	ERROR	159	■	RESUME
160	┐	OUT	161	■	ON
162	┐	OPEN	163	■	FIELD
164	┐	GET	165	┐	PUT
166	┐	CLOSE	167	┐	LOAD
168	■	MERGE	169	■	NAME
170	■	KILL	171	┐	LSET
172	┐	RSET	173	┐	SAVE
174	┐	SYSTEM	175	■	LPRINT
176	┐	DEF	177	■	POKE
178	┐	PRINT	179	┐	CONT
180	┐	LIST	181	┐	LLIST
182	┐	DELETE	183	⊗	AUTO
184	■	CLEAR	185	✕	CLOAD
186	┐	CSAVE	187	·	NEW
188	┐	TAB(189	·	TO
190	■	FN	191	·	USING
192	✕	VARPTR	193	≡	USR
194	+	ERL	195	✕	ERR
196	■	STRING\$	197	✕	INSTR
198	♥	CHECK	199	✕	TIME\$
200	◆	MEM	201	▲	INKEY\$
202	■	THEN	203	■	NOT
204	■	STEP	205	┐	+
206	┐	-	207	┐	+
208	┐	/	209	┐	[
210	■	AND	211	┐	OR
212	┐	>	213	┐	=
214	┐	<	215	≡	SGN
216	■	INT	217	┐	ABS
218	┐	FRE	219	┐	INP

220	⧻	POS	221		SQR
222	⧻	RND	223	/	LOG
224	▴	EXP	225		COS
226	▴	SIN	227	\	TAN
228	▴	ATN	229	◡	PEEK
230	●	CVI	231	⌫	CVS
232	↑	CVD	233	□	EOF
234	%. LOC		235	○	LOF
236	⌈	MKI\$	237	⌈	MKS\$
238	⌈	MKD\$	239	⌈	CINT
240	⌈	CSN6	241	⌈	COBL
242	◆	FIX	243	⌈	LEN
244	▲	STR\$	245	⌈	VAL
246	·	ASC	247	::	CHR\$
248	□	LEFT\$	249	::	RIGHT\$
250	⌘	MID\$	251	↓	'

Colour-Basic Doppeltokens

255 128	⌈	COLOUR	255 129	⌈	FCOLOUR
255 130	⌈	KEYPAD	255 131	⌈	JOY
255 132	⌈	PLOT	255 133	⌈	FGR
255 134	⌈	LGR	255 135	⌈	FCLS
255 136	⌈	PLAY	255 137	⌈	CIRCLE
255 138	⌈	SCALE	255 139	⌈	SHAPE
255 140	⌈	NSHAPE	255 141	⌈	XSHAPE
255 142	⌈	PAINT	255 143	⌈	CPOINT
255 144	⌈	NPLOT	255 145	⌈	SOUND
255 146	⌈	CHAR	255 147	⌈	RENUM
255 148	⌈	SWAP	255 149	⌈	FKEY
255 150	⌈	CALL	255 151	⌈	VERIFY
255 152	⌈	BGRD	255 153	⌈	NBGRD

(Guel19 fuer die neuen ROMs)

Anhang B: Dezimal-, Hexadezimal-, ASCII-Tabelle

0 = 00H	1 = 01H	2 = 02H
3 = 03H	4 = 04H	5 = 05H
6 = 06H	7 = 07H	8 = 08H
9 = 09H	10 = 0AH	11 = 0BH
12 = 0CH	13 = 0DH	14 = 0EH
15 = 0FH	16 = 10H	17 = 11H
18 = 12H	19 = 13H	20 = 14H
21 = 15H	22 = 16H	23 = 17H
24 = 18H	25 = 19H	26 = 1AH
27 = 1BH	28 = 1CH	29 = 1DH
30 = 1EH	31 = 1FH	32 = 20H
33 = 21H	34 = 22H	35 = 23H
36 = 24H	37 = 25H	38 = 26H
39 = 27H	40 = 28H	41 = 29H
42 = 2AH	43 = 2BH	44 = 2CH
45 = 2DH	46 = 2EH	47 = 2FH
48 = 30H	49 = 31H	50 = 32H
51 = 33H	52 = 34H	53 = 35H
54 = 36H	55 = 37H	56 = 38H
57 = 39H	58 = 3AH	59 = 3BH
60 = 3CH	61 = 3DH	62 = 3EH
63 = 3FH	64 = 40H	65 = 41H
66 = 42H	67 = 43H	68 = 44H
69 = 45H	70 = 46H	71 = 47H
72 = 48H	73 = 49H	74 = 4AH
75 = 4BH	76 = 4CH	77 = 4DH
78 = 4EH	79 = 4FH	80 = 50H
81 = 51H	82 = 52H	83 = 53H
84 = 54H	85 = 55H	86 = 56H
87 = 57H	88 = 58H	89 = 59H
90 = 5AH	91 = 5BH	92 = 5CH
93 = 5DH	94 = 5EH	95 = 5FH
96 = 60H	97 = 61H	98 = 62H
99 = 63H	100 = 64H	101 = 65H
102 = 66H	103 = 67H	104 = 68H
105 = 69H	106 = 6AH	107 = 6BH
108 = 6CH	109 = 6DH	110 = 6EH
111 = 6FH	112 = 70H	113 = 71H
114 = 72H	115 = 73H	116 = 74H
117 = 75H	118 = 76H	119 = 77H
120 = 78H	121 = 79H	122 = 7AH
123 = 7BH	124 = 7CH	125 = 7DH
126 = 7EH	127 = 7FH	128 = 80H
129 = 81H	130 = 82H	131 = 83H
132 = 84H	133 = 85H	134 = 86H
135 = 87H	136 = 88H	137 = 89H

138	=	8AH		139	=	8BH	-	140	=	8CH	-
141	=	80H	□	142	=	8EH	-	143	=	8FH	-
144	=	90H	■	145	=	91H	■	146	=	92H	-
147	=	93H	■	148	=	94H	■	149	=	95H	-
150	=	96H	■	151	=	97H	■	152	=	98H	-
153	=	99H	■	154	=	9AH	■	155	=	9BH	-
156	=	9CH	■	157	=	9DH	■	158	=	9EH	-
159	=	9FH	■	160	=	A0H	■	161	=	A1H	-
162	=	A2H	■	163	=	A3H	■	164	=	A4H	-
165	=	A5H	■	166	=	A6H	■	167	=	A7H	-
168	=	A8H	■	169	=	A9H	■	170	=	AAH	-
171	=	ABH	■	172	=	ACH	■	173	=	ADH	-
174	=	AEH	■	175	=	AFH	■	176	=	B0H	-
177	=	B1H	■	178	=	B2H	■	179	=	B3H	-
180	=	B4H	■	181	=	B5H	■	182	=	B6H	-
183	=	B7H	■	184	=	B8H	■	185	=	B9H	-
186	=	BAH	■	187	=	BBH	■	188	=	BCH	-
189	=	B0H	■	190	=	BEH	■	191	=	BFH	-
192	=	C0H	■	193	=	C1H	■	194	=	C2H	-
195	=	C3H	■	196	=	C4H	■	197	=	C5H	-
198	=	C6H	■	199	=	C7H	■	200	=	C8H	-
201	=	C9H	■	202	=	CAH	■	203	=	CBH	-
204	=	COH	■	205	=	CDH	■	206	=	CEH	-
207	=	CFH	■	208	=	D0H	■	209	=	D1H	-
210	=	D2H	■	211	=	D3H	■	212	=	D4H	-
213	=	D5H	■	214	=	D6H	■	215	=	D7H	-
216	=	D8H	■	217	=	D9H	■	218	=	DAH	-
219	=	DBH	■	220	=	DCH	■	221	=	D0H	-
222	=	DEH	■	223	=	DFH	■	224	=	E0H	-
225	=	E1H	■	226	=	E2H	■	227	=	E3H	-
228	=	E4H	■	229	=	E5H	■	230	=	E6H	-
231	=	E7H	■	232	=	E8H	■	233	=	E9H	-
234	=	EAH	■	235	=	EBH	■	236	=	ECH	-
237	=	EDH	■	238	=	EEH	■	239	=	EFH	-
240	=	F0H	■	241	=	F1H	■	242	=	F2H	-
243	=	F3H	■	244	=	F4H	■	245	=	F5H	-
246	=	F6H	■	247	=	F7H	■	248	=	F8H	-
249	=	F9H	■	250	=	FAH	■	251	=	FBH	-
252	=	FCH	■	253	=	FDH	■	254	=	FEH	-
255	=	FFH	■								

Anhang C Der Farbspeicher

Vielleicht ist Ihnen auch schon aufgefallen, daß man beim POKEN in den Farbspeicher bei gleichem Argument andere Farben bekommt als mit dem COLOUR-Befehl.

Folgende Tabelle gibt Ihnen an, wie die einzelnen Farben abgespeichert werden.

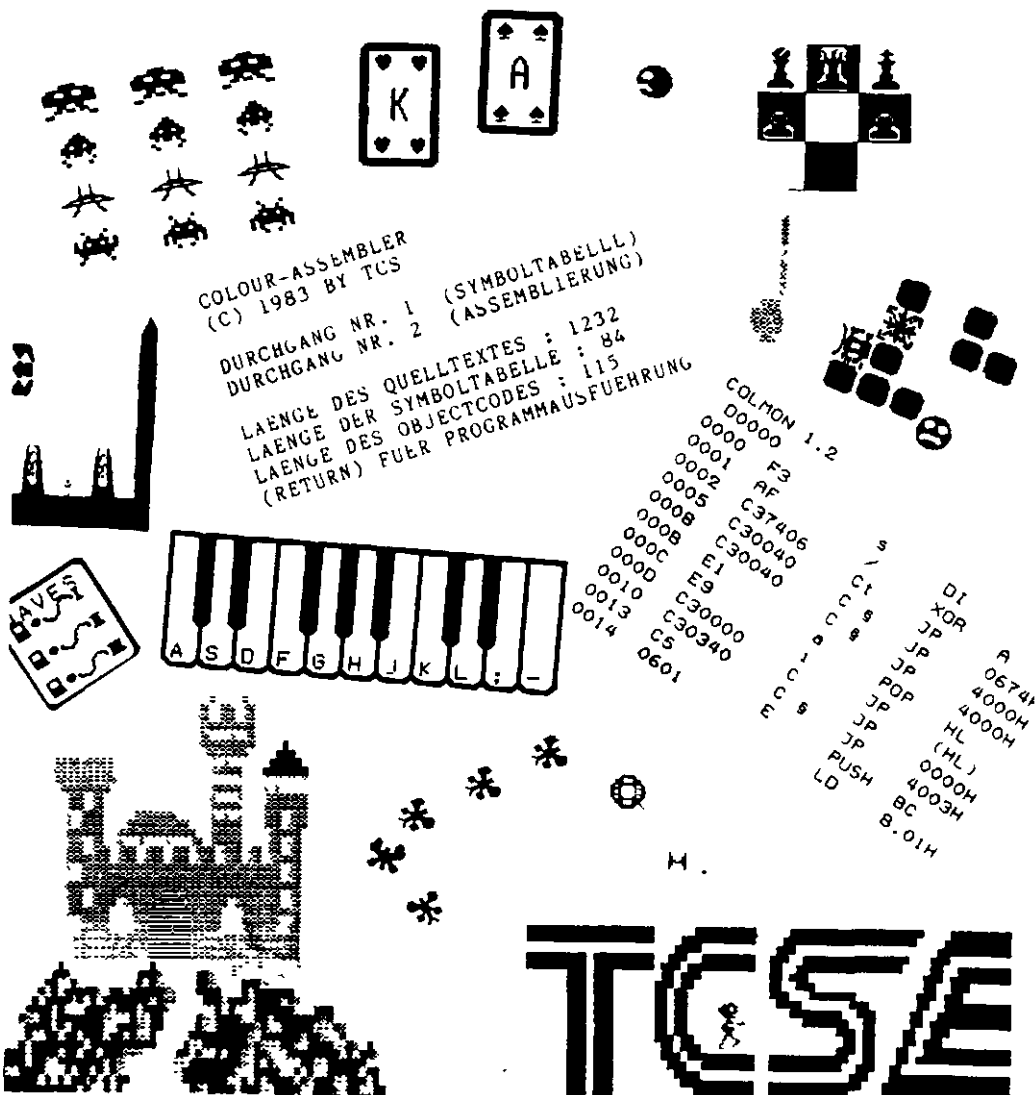
(Hinweis: Der Farbspeicher hat nur 4 Bit pro Adresse, d.h. wenn Sie den Farbspeicher mit dem PEEK-Befehl abfragen müssen Sie das Ergebnis mit 15 'undieren', um die 4 nicht belegten Bits zu unterdrücken. Z.B.:
X=PEEK(&HF000) AND 15

Es folgt nun die Vorbelegung der Farbtabelle ab &H4390 :

Farbe	Bit 3	Bit 2	Bit 1	Bit 0	Wert	Farb-Nr.
Grau	0	0	0	0	0	10
Cyan	0	0	0	1	1	7
Rot	0	0	1	0	2	3
Weiß	0	0	1	1	3	1
Gelb	0	1	0	0	4	4
Grün	0	1	0	1	5	2
Orange	0	1	1	0	6	5
Hellgelb	0	1	1	1	7	11
Blau	1	0	0	0	8	6
Hellblau	1	0	0	1	9	9
Rotviolett	1	0	1	0	10	15
Blauviolett	1	0	1	1	11	12
Hellgrau	1	1	0	0	12	13
Türkis	1	1	0	1	13	14
Magenta	1	1	1	0	14	8
Hellweiß	1	1	1	1	15	16

Colour-Genie

Software



Verzeichnis der verfügbaren Colour-Genie-Software

1.) HILFSPROGRAMME, PROGRAMMIERSPRACHEN

COLOUR-COMPILER: Dieses Programm ist extrem wertvoll. Es bietet die Möglichkeit, Basicprogramme in Maschinenspracheprogramme zu übersetzen.

Der wesentliche Vorteil: Ihr Programm läuft nach Compilierung c.a. 40 mal (!) schneller.

Dies geschieht interaktiv, d.h. Basicprogramm, Maschinenprogramm und Compiler stehen gleichzeitig im Speicher, was ein sehr schnelles Arbeiten ermöglicht. Dabei können c.a. 9.5 kByte Basicprogramm verarbeitet werden. Verzichtet man auf die hochauflösende Grafik, steht noch mehr Speicher zur Verfügung. Voraussetzung: Ihr COLOUR-GENIE MUSS AUF 32K RAM ERWEITERT SEIN. Die meisten Colour-Basic-Befehle werden vom Colour-Compiler unterstützt. Die wesentlichste Einschränkung ist, daß der Colour-Compiler nur Ganzzahlen (Integers) verarbeitet.

Preis: 69.- DM

COLOUR-MONITOR I: Ein Maschinensprachemonitor mit starkem Befehlssatz, z.B. Disassemblieren, Ascii/Hex-Dump, Bänder laden/schreiben, Bytes suchen, Speicher editieren/verschieben/relozieren, Hex/Dezimal-Umwandlung u.a.m.

Preis: 39.- DM

COLOUR-ASSEMBLER: Mit diesem Programm können Maschinenspracheprogramme auf komfortable Weise entwickelt werden. Das Programmieren geschieht mit 280-Mnemonics, Labels etc. Der Assembler erzeugt dann das Maschinenspracheprogramm, das zur Probe auch direkt ausgeführt werden kann.

Eine weitere Besonderheit des Assemblers ist, daß der Assembler-Text im ganz normalen Basic-Modus erzeugt wird, so daß alle Basic-Befehle wie EDIT, LIST, CSAVE vom Assembler ausgenutzt werden.

Preis: 69.- DM

SOUND-EDITOR (auch im Handbuch "Colour Basic leicht gelernt" aufgelistet): Ein nützliches Hilfsprogramm zur Programmierung des PSG-ICs (Sound-Chip). Sie editieren die PSG-Register auf dem Bildschirm und der entsprechende Ton wird gleichzeitig ausgegeben. Abschließend gibt das Programm alle PSG-Registerinhalte in dezimaler Schreibweise aus.

Preis: 25.- DM

ZEICHEN-EDITOR (auch im Handbuch "Colour Basic leicht gelernt" aufgelistet): Dieses Programm ermöglicht es, 64 der 128 frei definierbaren Zeichen auf dem Bildschirm übersichtlich zu editieren. Es können auch Grafiken, wie z.B. eine Schreibschrift, erstellt werden. Abschließend werden die definierten Zeichen in ein Basicprogramm geschrieben, was diese für den späteren Gebrauch oder zur Entwicklung eigener Programme wieder definiert. So wird die unständliche Handhabung der Programmierung der definierbaren Zeichen umgangen.

Preis: 25.- DM

GRAFIK-EDITOR: Wenn Sie Ihr Colour-Genie auf 32K RAM aufgerüstet haben, können Sie dieses Programm einsetzen, denn der "Grafik-Editor" hat eine Länge von 26000 Bytes! Er bietet die Möglichkeit, Grafiken im FGA-Modus mit einer Vielzahl von leistungsstarken Kommandos zu kreieren. Abschließend wird ein Basic-Programm erzeugt werden, das, eingebunden in ein eigenes Programm, das Bild blitzschnell wieder auf den Bildschirm bringt. Sie können also auch komplexe Grafiken sehr schnell fertigstellen und sie hinterher beliebig verwenden.
Preis: 69.- DM

SHAPER: Die Handhabung des Basicbefehls "SHAPE" ist recht umständlich. Hier bringt der "Shaper" Abhilfe: Mit wenigen Tastendruckern erzeugen Sie eine Figur, die abschließend im Format der "Shape-Table" abgespeichert wird.
Preis: 25.- DM

COLROT: Ein sehr nützliches Grafikprogramm, für alle die mit großen Texten Aufmerksamkeit erregen wollen (z.B. im Schaufenster). Colrot erzeugt Laufschriften mit bis zu 512 Zeichen Länge, Titel und Fußschrift, Inversdarstellung, Intermezzo u.v.a.m. Neu: Texte können auf Band gespeichert werden. Ein komfortabler Editor ermöglicht einfaches Arbeiten.
Preis: 69.- DM

BASICODE 2 (11): Der neue Basicode-Standard für das Colour-Genie (kompatibel zu Basicode 1). Dieses Programm macht Ihr Colour-Genie Basicode kompatibel! Basicode ist das Standard-Basic, in dem z.B. das WDR-Fernsehen Programme austrahlt. Eine Hardwareänderung ist nicht nötig.
Preis: 25.- DM

ZEICHENEDITOR+: Dieser Editor, in Maschinensprache geschrieben, ermöglicht es alle 128 Zeichen sehr schnell und komfortabel zu editieren. Dabei gibt es sogar Befehle um Kreise zu zeichnen, Zeichen zu kopieren/duplizieren/rotieren/invertieren u.v.a.m.
Preis: 39.- DM

BASIC+5: Dieses Programm erweitert das normale Basic um 5 Befehle: Im Grafikmodus können beliebige Texte dargestellt werden, Rechteckflächen können gemalt werden, mit "SAVE" können Maschinenspracheprogramme gesichert und mit "LOAD" komfortabel geladen werden. Ferner steht ein Eingabebefehl zur Verfügung, der an beliebige Speicherzellen schreibt.
Preis: 39.- DM

ZEICHENSATZ: Dieses Programm läßt Sie den vorhandenen Zeichensatz für alle ASCII-Zeichen wahlweise gegen einen von acht neuen, interessanten Zeichensätzen ersetzen.
Für dieses Programm muß Ihr Colour-Genie mit 32K RAM ausgerüstet sein.
Preis: 25.- DM

SCREEN-PRINTER: (Auch in dem Buch "Das Colour-Genie-Buch 1" aufgelistet.) Auf dieses Programm haben die Besitzer des STAR-Druckers DP 510 bzw. DP 515 sicher gewartet!
Es ermöglicht es Ihnen, den Bildschirm jederzeit komplett auf Ihren Drucker auszugeben, egal ob Sie im FGR- oder im LGR-Modus sind. Definierte und feste Grafikzeichen werden ebenfalls mit ausgedruckt. Auch für dieses Programm brauchen Sie 32K RAM und natürlich einen STAR DP 510 / DP 515 Drucker.
Preis: 39.- DM

COLOUR-MONITOR III: Dieses Programm wird alle, die auf Maschinensprache-Ebene programmieren begeistern!
Der "Colour-Monitor III" ist nicht nur ein Monitor mit den üblichen Befehlen, wie ASCII-Dump, Hex-Dump, Disassemblieren, Bänder schreiben etc., sondern ein Monitor mit eingebautem Assembler, Find-Assembler und der Möglichkeit Maschinenspracheprogramme in Einzelschrittausführung zu testen!
Das Arbeiten wird neben dem Super-Befehlssatz durch raffinierte Aufteilung des Bildschirms erleichtert.
Für dieses Programm benötigen Sie ein Colour-Genie mit 32K Ram.
Ein ausführliches deutsches Handbuch liegt bei.
Preis: 95.- DM

COLOUR-FORTH: Endlich gibt es eine leistungsstarke Version der Forth-Programmiersprache auch für das Colour-Genie!
"Colour-Forth" ist eine für Kassettenbetrieb angepaßte Version des FIG-Forth, die um viele Befehle für das Colour-Genie erweitert wurde (Graphik, Ton...). Der wesentliche Vorteil der Programmiersprache Forth ist die extrem schnelle Ausführungszeit der Programme, die in unserem Forth noch durch Benutzung des eingebauten Assemblers gesteigert werden kann!
Für dieses Programm benötigen Sie 32K Ram, sowie die neuen Basic-Roms, die seit April '83 in alle Colour-Genies eingebaut sind.
Ein ausführliches deutsches Handbuch wird mitgeliefert.
Preis: 95.- DM

COLOUR-TEXT: Endlich können alle Colour-Genie-Besitzer, die einen Drucker haben in den Genuß einer komfortablen Textverarbeitung kommen!
Colour-Text erweitert die Bildschirmbreite auf 55 Zeichen, die maximale Zeilenlänge ist 64 Zeichen (horizontales Scrolling). Sie sind mit dem Cursor ständig im Text, und keine Steuerzeichen verunstalten den Text, da die verschiedenen Schriftarten durch Farben gekennzeichnet sind.
(Für Benutzer eines Schwarz/Weiss-Sichtgerätes extra Statuszeile.) Das Programm unterstützt alle gängigen Drucker (STAR, EPSON, ITOH, Typenrad). Wenn Sie einen grafikfähigen STAR-, EPSON- oder ITOH-Drucker haben können Sie sogar selbstdefinierte Grafikzeichen in den Text einfügen!
Ein deutsches Handbuch liegt bei.
Preis: 95.- DM

2.) SPIEL- u. GRAFIKPROGRAMME

INVASION AUS DEM WELTRAUM: Ein Maschinenspracheprogramm, das die Sound- u. Grafikmöglichkeiten des Colour-Genies voll ausnutzt. Ihre Aufgabe ist es, einen Pulk von Invasoren abzuwehren, der sich der Erde nähert. Die Invasoren werden dabei immer schneller und gefährlicher. Preis: 39.- DM

PUNKTEJAGD: Bei diesem Spiel geht es darum, alle Punkte in einem Labyrinth aufzusammeln, bevor Sie von einem Wächter eingeholt werden. Schnelle Grafik und Ton durch Maschinensprache. Preis: 25.- DM

WURM: Unser derzeit schnellstes Action-Spiel. Ein Wurm kommt von oben den Bildschirm herab und versucht Sie zu vernichten. Dabei hat er die Spinne, die Fliege und die Ente als Helfer. Kein Spiel für ruhige Stunden! Preis: 39.- DM

BREAK OUT: Bei diesem Video-Spiel muß eine Mauer mit Ihrem Ball zerstört werden. Dabei können verschiedenen Schwierigkeitsgrade vorgewählt werden. Hohe Geschwindigkeit durch Maschinenspracheprogrammierung. Preis: 39.- DM

DEMOPROGRAMM: Hiermit können Sie besser als mit der mit Ihrem Colour-Genie z.Zt. mitgelieferten englischen Demokassette die Möglichkeiten Ihres Colour-Genies demonstrieren. Preis: 25.- DM

ANDROMEDA: Ein erstklassiges dreidimensionales Weltraumspiel mit ansprechender Grafik. Feindliche Raumschiffe kommen auf Sie zu; vernichten Sie diese, bevor sie Ihnen wertvolle Energie abgesaugt haben. Ist die Zeit abgelaufen, so eilt Ihnen Ihre Mutterbasis zu Hilfe; aber das Auftanken will auch gelernt sein. Preis: 39.- DM

MAU-MAU: Endlich haben Sie einen Spielpartner, der nicht wutend die Karten wegwirft, wenn er am verlieren ist. Dies dürfte wohl aber auch selten der Fall sein, vielmehr wahrscheinlich ist es, daß Sie den Computer vor Wut ausschalten, da ihr Computer Dank einer hervorragenden Taktik auf Sieg programmiert ist. (Er schummelt nicht !!) Gespielt wird nach den üblichen Regeln. Preis: 25.- DM

HEKTIK: Stellen Sie sich vor, Sie wären in einem Neubau mit 6 Geschossen, bei dem die Ebenen durch Leitern verbunden sind. Jetzt kommen Ihnen von oben Verfolger entgegen. Sie müssen sich ihnen stellen, denn es gibt keinen Fluchtweg. Graben Sie an strategisch wichtigen Stellen Löcher in den Boden, um so Ihre Verfolger auszuschalten. Preis: 39.- DM

METEOR: Ein Super-Action-Spiel im Grafik-Modus Ihres ColourGenies. Sie befinden sich in einem Meteoritenfeld und werden von Meteoriten, Sternen und schiessenden Ufos bedrängt. Das Programm zeichnet sich durch sehr schnelle Grafik und gute Toneffekte aus.
Preis: 69.- DM

MOTTEN: Bei diesem Videospiel müssen Sie Kolonnen von Motten bekämpfen, die in gefährlichen Sturzflügen anfliegen und dabei auch noch schießen.
Preis: 39.- DM

PANIK: Ein Programm für 2 Spieler. Man muß versuchen, den Gegner einzumauern. Wählbare Geschwindigkeit macht "Mauer" entweder zu einem Reaktions- oder zu einem Strategiespiel. Das Programm kann wahlweise mit Joysticks oder mit der Tastatur bedient werden.
Preis: 25.- DM

TAUSENDFUß: Hier sollen Sie einen Wurm so steuern, daß er nur Futter aber kein Gift frißt. Dazu kommen noch Kraftfutter und Gegengift. Ein sehr unterhaltsames Geschicklichkeitsspiel für die ganze Familie. Bei Spielbeginn können viele verschiedene Geschwindigkeiten und Schwierigkeitsstufen vorgewählt werden.
Preis: 39.- DM

COLOUR-SCHACH: Jetzt können Sie gegen Ihr Colour-Genie auch Schach spielen. Colour-Schach bietet 4 verschiedene Spielstärken. Aufzeichnung eines laufenden Spieles auf Kassette, Ändern von Stellungen, einen Demonstrationsmodus und anderes mehr. Die Figuren werden auf ansprechende Weise grafisch dargestellt.
Preis: 69.- DM

EXNIMROID: Ein Denkspiel, abgeleitet vom bekannten Nim-Spiel. Es geht darum, aus mehreren Häufchen den letzten Spielstein zu nehmen. Sie spielen gegen den Computer.
Preis: 25.- DM

KINGS: Eine Regierungssimulation, bei der Sie für eine Dauer von 8 Jahren eine Insel regieren sollen. Dabei kann nur der kluge Einsatz aller Faktoren zu einem Erfolg führen. Haben Sie Ihre Amtszeit überlebt, werden Ihre Taten ausgewertet, und Sie können dann auch weiter regieren.
Preis: 25.- DM

CONQUER: Ein spannendes Videospiel - erobern Sie die Spinnenfestung!
Preis: 25.- DM

MAMPFMAN: Ähnlich wie unser Spiel "Punktejagd", nur daß Sie hier von mehreren Gespenstern verfolgt werden, die Sie unter bestimmten Bedingungen jedoch auch fressen können.
Preis: 25.- DM

MAMPFMAN 2: Die neue Version von Mampfman ! Excellente Grafik und hohe Spieldynamik. Ein sehr empfehlenswertes Programm !
Preis: 39.- DM

COLOUR-KONG: Das absolute Videospiel ! Retten Sie Ihre Freundin vor dem wilden Affen. Mehrere Ebenen, super Grafik und Musik. Auf dem Band finden Sie Versionen für 32K und 16K RAM
Preis: 69.- DM

PANZERKAMPF: Ein Spiel für zwei Personen (JOYSTICKS und 32K RAM erforderlich). Liefern Sie sich mit Ihrem Gegner spannende Panzerschlachten ! Das Programm verfügt über 3 verschiedene Gelände mit Minen etc.
Preis: 69.- DM

EIS: Ein spannendes, zugleich jedoch auch recht schwieriges Actionspiel mit strategischen Elementen. Sie befinden sich in einem Kuhlraum und müßen Eisblöcke so ans Rutschen bringen, daß die gefährlichen Schneemonster vernichtet werden.
Preis: 39.- DM

EAGLE: Vernichten Sie verschiedene Schwadronen von Vögeln, bevor Sie versuchen das große Mutterschiff zu zerstören.
Preis: 39.- DM

GAME of LIFE: Dieses bekannte Programm simuliert das Wachstum von Bakterienkulturen. Die Regeln werden im Programm erklärt und einige interessante Figuren sind fest abgespeichert.
Preis: 25.- DM

SAUG: Ein lustiges Videospiel, bei dem Sie aus einem unterirdischen Labyrinth Termiten ansaugen müßen, wobei Sie sich vor giftigen, roten Termiten hüten müßen.
Das Spiel kann auch mit Joysticks gesteuert werden.
Preis: 39.- DM

EXREVERSIC: Spielen Sie Reversi gegen Ihr Colour-Genie ! Dabei kann zwischen mehreren Spielstufen gewählt werden. Das Spielbrett wird auf ansprechende Weise graphisch dargestellt.
Preis: 39.- DM

BANG-BANG: Zwei Cowboys begegnen sich in der Prarie und liefern sich ein erbittertes Duell. Für dieses Spiel sind Joysticks erforderlich.
Preis: 39.- DM

NETZO: Versuchen Sie mit Ihrem Pinsel alle Flächen auszumalen, ohne von Ihren Gegnern erwischt zu werden.
Preis: 39.- DM

DEATH-TRAP, ein dreidimensionales Grafik-Abenteuerspiel (Adventure):

Ein Programm zum Wahnsinnig werden! Sie befinden sich in einem Labyrinth mit über 1100 Räumen, in dem es von gefährlichen Einwohnern und Gegenständen wimmelt. Finden Sie die wichtigen Gegenstände und wenden Sie sie richtig an, um aus dem Labyrinth zu entkommen. Zwei Voraussetzungen müssen allerdings erfüllt sein: Ihr Colour-Genie muß 32K RAM haben, und Sie müssen elementare Englischkenntnisse besitzen, da die Sie Kommandos in Form von englischen Sätzen eingeben.

Preis: 69.- DM

TCS-CHOPPER: Die Geschmäcker sind verschieden - aber dies ist wohl das beste Videospiel, das es z.Zt. für das Colour-Genie gibt! Dreizehn Ihrer Kameraden sind in einem von vier riesigen Labyrinthen ausgesetzt worden. Versuchen Sie nun, diese mit Ihrem Hubschrauber zu retten! Aber Vorsicht, Ihre Freunde werden scharf bewacht!

32K RAM sind auch hier notwendig.

Preis: 69.- DM

DOPPEL-WURM: Vielleicht kennen Sie unser beliebtes Spiel "Tausendfuß". "Doppel-Wurm" ist sehr ähnlich, allerdings spielen hier zwei Spieler gegeneinander, was die Spannung natürlich erhöht. Achtung: Nur für Joystick-Besitzer!

Preis: 39.- DM

MADTREE: Ein schwieriges Videospiel: Sie sind eine fleißige Biene und müssen eine Reihe von Blumen ständig bestäuben, damit diese nicht eingehen. Doch diese Blumen sind undankbar und gefährden Sie.

Preis: 39.- DM

DIG-BOY: In Ihrem unterirdischen Reich kämpfen Sie gegen Drachen und andere Untiere. Locken Sie diese unter einen der wackligen Felsen, um sie zu zerschmettern.

Dieses Spiel ist sehr empfehlenswert, da es eine Menge Abwechslung bietet.

Preis: 39.- DM

EMPIRE: Für alle, die nicht nur reine Aktion wollen, ist "Empire" das ideale Spiel. Dieses Spiel kann mit bis zu sechs Spielern gleichzeitig gespielt werden, der Computer spielt ebenfalls mit. Jeder Spieler hat ein kleines Reich, das nun durch Handel, Politik, Krieg u.s.w. zu Wohlstand kommen soll. Dabei stehen Sie mit den anderen Nationen ständig in Beziehung, sei es durch Handel oder durch Krieg.

Ansprechende Grafik macht das Programm noch interessanter. Ihr Colour-Genie muß für "Empire" auf 32K RAM aufgerüstet sein.

Preis: 69.- DM

LUNAR-LANDER: Eine perfekte Mondlandungssimulation mit Supergrafik ! Die Idee ist zwar alt, aber die Ausführung dieses Programmes macht den "Lunar-Lander" zu einem unserer besten Spiele. Sie benötigen 32K Ram. Das Spiel kann wahlweise auch mit den EG2013-Joysticks gesteuert werden.
Preis: 39.- DM

COLOUR-FROGGER: Retten Sie Ihren Frosch zuerst über die Straße und dann über den Fluß ! Ein spannendes Unterhaltungsspiel, an dem die ganze Familie lange Zeit Spass haben wird. Besonders gelungen ist bei diesem Programm die musikalische Untermalung.
Preis: 39.- DM

CRAZY-PAINT: Das neueste Spiel von J. Buchmüller, dem Author des TCS-Choppers - Super-Grafik und Super-Ton ! In 16 Levels müssen Sie, ähnlich wie bei unserem Spiel 'Netzo', versuchen alle Flächen auszumalen, ohne sich von den hektischen Gegnern erwischen zu lassen.
Preis: 69.- DM

MEMORY: Das beliebte Brettspiel nun auch als Version für das Colour-Genie ! Schöne Grafik und die gute Spielidee machen dieses Programm empfehlenswert.
Preis: 39.- DM

3.) LEARNPROGRAMME, ELEKTRONIKPROGRAMME:

MATHEMATIK-LEARNPROGRAMM: Interessant für Kinder bis in's Grundschulalter. Wahlweise können die vier Grundrechenarten geübt werden, wobei man zwischen verschiedenen Schwierigkeitsgraden wählen kann. Lustige grafische Gestaltung sorgt dafür, daß das Üben Spaß macht.
Preis: 25.- DM

GRUNDEIGENSCHAFTEN DES LICHTS (OPTIK): Diese 32KByte lange Programm erklärt die Brechung, Reflexion, verschiedene Linsentypen... Ansprechende Grafik, Benutzereingaben, Menu und Übungsaufgaben machen dieses Programm sehr lehrreich.
Preis: 39.- DM

NE555: Erklärung und Anwendung des Timer-ICs NE555 werden mit diesem Programm zum Kinderspiel. Tolle Grafik und ausführliche Dokumentation heben dieses Programm hervor.
Preis: 25.- DM

PLOTTER: Plotter ist ein anspruchsvolles Mathematikprogramm, das beliebige Funktionen zeichnet, Wertetabellen erstellt ...
Preis: 39.- DM

PHYSIK DES TRANSISTORS: Ähnlich aufwendig wie beim Programm 'Optik' werden Sie hier in die Grundeigenschaften des Transistors, verschiedene Schaltungen, Kennlinien u.s.w. eingewiesen.
Für dieses Programm sind 32K Byte RAM erforderlich.
Preis: 39.- DM

4.) MUSIKPROGRAMME

MUSIK: Dieses Programm spielt festprogrammierte Musikstücke mit verbluffender Qualität.

Preis: 25.- DM

ORGEL: Spielen Sie Orgel auf Ihrem Colour-Genie ! Ihnen stehen zwei Manuale, einstellbares Delay und sogar eine Schlagzeugbegleitung zur Seite.

Preis: 25.- DM

COLOUR-SYNTHESIZER: Ein Programm, das man gesehen haben muß, um es überhaupt für möglich zu halten. Der Colour-Synthesizer macht aus Ihrem Colour-Genie einen vollwertigen 3-Kanal Synthesizer mit VCO, VCA, Hüllkurve, Schlagzeug...

Acht Einstellungen aller Regler können vorprogrammiert, auf Tastendruck abgerufen und auf Band gespeichert werden. Verbluffend ist auch die graphische Darstellung.

Preis: 69.- DM

COMPOSER: Mit diesem Programm können Sie sehr einfach eigene Musikstücke komponieren. Noten, Pausenzeichen u.s.w. werden grafisch dargestellt, können editiert und auf Band gespeichert werden.

Preis: 39.- DM

**** STAND DER TCS-SOFTWARELISTE, COLOUR-GENIE, VOM 18.12.1983 ****

Beachten Sie auch die für das Colour-Genie lieferbaren Bücher:

Technisches-Handbuch: Enthält die kompletten Schaltpläne, Erklärungen, CRTC-Unterlagen, PSG-Unterlagen...

Preis: 49.- DM

Dokumentiertes ROM-Listing für das Colour-Genie: Unerläßlich für den Maschinensprache-Programmierer. In diesem Buch ist der komplette ROM und reservierte RAM aufgelistet und erklärt.

Preis: 59.- DM

DAS Colour-Genie BUCH I: Auf über 100 Seiten finden Sie hier interessante Programme - von Spielen bis zu Hilfsprogrammen, zum Teil auch in Maschinensprache - sowie wertvolle Tricks und Erklärungen. Ein Muß für jeden Colour-Genie-Besitzer!

Preis: 39.- DM

In Vorbereitung: Software für Diskette, neue Spiele ...

5.) PROGRAMME FÜR DISKETTENBENUTZER

TAPE-DISK-CONVERTER: Dieses Programm ermöglicht es Ihnen, Ihre Maschinensprache-Kassetten (sofern ungeschützt) auf Diskette zu überspielen, wobei eine Verschiebung im Speicher möglich ist. Außerdem kann man von Diskette auf Diskette, von Kassette auf Kassette und von Diskette auf Kassette überspielen.

Preis: 45.- DM

COLZAP: Mit "Colzap" können Sie auf jeder (ungeschützten) Colour-Genie-Diskette alle Sektoren lesen, beliebig ändern (in ASCII oder Hex), Sektoren schreiben, den Disknamen anzeigen, den freien Platz auf der Diskette ausgeben lassen, u.ä.m.

Preis: 45.- DM

BESTELLFORMULAR

Zur Bestellung von Colour-Genie-Programmen können Sie dieses Formular benutzen. Schicken Sie dieses an Ihren GENIE-Handler oder direkt an

TCS Computer GmbH
Kohnstraße 4
5205 St. Augustin 2
Tel.: 02241 / 28071

Bankverbindung:

Raiffeisenbank Menden BLZ 37069507 Konto-Nr. 1525

Der Versand erfolgt per Nachnahme oder Vorkasse auf obiges Konto (dies natürlich nicht, wenn Sie bei Ihrem Handler bestellen).

Ich bitte um Zusendung folgender TCS Colour-Genie-Programme:

Anzahl	:	Name des Programms	:	Preis/DM	:	Gesamt
-----	-----	-----	-----	-----	-----	-----
:	:	:	:	:	:	:
-----	-----	-----	-----	-----	-----	-----
:	:	:	:	:	:	:
-----	-----	-----	-----	-----	-----	-----
:	:	:	:	:	:	:
-----	-----	-----	-----	-----	-----	-----
:	:	:	:	:	:	:
-----	-----	-----	-----	-----	-----	-----
:	:	:	:	:	:	:
-----	-----	-----	-----	-----	-----	-----
:	:	:	:	:	:	:
-----	-----	-----	-----	-----	-----	-----
Gesamt:					DM	

Meine Adresse ist:

Name:

Straße:

Ort:

Datum: Unterschrift:

Raum für Notizen